



ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ

DEPARTMENT OF INFORMATICS

Πτυχιακή εργασία

Students Absence - Attendance taking application

Students Absence - Εφαρμογή παρουσιολόγιου

Όνομα: **Ευστράτιος Δερμανούτσος**

Αριθμός Μητρώου: Π19041

Email: stratis.dermanoutsos@gmail.com

Επιβλέπων καθηγητής: **Αλέξης Ευθύμιος**

Περιεχόμενα

Περιεχόμενα	2
Περίληψη	3
Εισαγωγή	3
Η εφαρμογή	3
Οι ανάγκες που εξυπηρετεί	4
Ανασκόπηση πεδίου	4
Απουσιολόγιο	5
Η ιδέα	6
Εγκατάσταση της εφαρμογής	7
Εγχειρίδιο χρήστη / User manual	17
Εισαγωγική περιγραφή	18
Κεντρικό μενού	18
Επιλογές χρήστη	20
Πρώτη εκκίνηση	21
Χρήση από το κοινό	23
Επιπλέον χαρακτηριστικά	28
Αρχιτεκτονική συστήματος	31
Παρουσίαση βάσης δεδομένων	31
Παρουσίαση backend	33
Παρουσίαση frontend	34
Μελλοντικές επεκτάσεις	35
Βιβλιογραφία	36
Λόγος δημιουργίας	37

Περίληψη

Η εφαρμογή που δημιούργησα είναι μια ολοκληρωμένη πλατφόρμα εισαγωγής και διαχείρισης απουσιών μαθητών, με την ονομασία “Students Absence”. Η εφαρμογή προσφέρει μια ικανοποιητική και απρόσκοπτη εμπειρία καταγραφής απουσιών σε καθηγητές και χρήστες. Μέσω της “Students Absence”, οι χρήστες μπορούν να περιηγηθούν άμεσα στην τάξη και λίστα μαθητών για την διδακτική ώρα ενδιαφέροντος, να προσθέσουν αλλά και να διορθώσουν απουσίες και να καταχωρίσουν τις εγγραφές μαζικά στο τέλος της διάλεξης. Η εφαρμογή υιοθετεί το μοντέλο αρχιτεκτονικής Minimal API της .NET, προσφέροντας μια οργανωμένη και εύκολα κατανοητή δομή κώδικα, καθώς και δυνατότητες αυτόματης ενημέρωσης του χρήστη για αλλαγές στα δεδομένα. Με την “Students Absence”, οι χρήστες απολαμβάνουν την εύκολη ολοκλήρωση μιας πάγιας και υποχρεωτικής διαδικασίας, ενώ εγώ επιτυγχάνω τη δημιουργία μιας πλήρους πλατφόρμας με σκοπό την εξέλιξη της σε startup.

Εισαγωγή

Η εφαρμογή

Η παρούσα εργασία Students Absence εξετάζει τη δημιουργία ενός πρωτοπόρου ηλεκτρονικού απουσιολογίου που εστιάζει στην ευκολία της χρήσης από καθηγητές όλων των ηλικιών. Εξετάζει τη διαδικασία ανάπτυξής του από την αρχή μέχρι το

τέλος, φωτίζοντας τον σχεδιασμό του, τα χαρακτηριστικά του και την ευκολία χρήσης για τους χρήστες.

Οι ανάγκες που εξυπηρετεί

Η εφαρμογή παρέχει τη δυνατότητα εισαγωγής απουσιών. Κάθε χρήστης (καθηγητής) έχει τη δυνατότητα να επιλέξει το τμήμα και το μάθημα που διδάσκει, να έχει άμεση πρόσβαση στη λίστα μαθητών που αντιστοιχούν στα παραπάνω στοιχεία και να επεξεργαστεί τις απουσίες που τον/-ην αφορούν. Στο τέλος της διαδικασίας, για μέγιστη ασφάλεια, οι απουσίες οριστικοποιούνται μόνο με τη χρήση του μοναδικού κωδικού ΠΙΝ του εκάστοτε καθηγητή.

Ανασκόπηση πεδίου

Η καταχώρηση των απουσιών στα σχολεία αποτελεί διαδικασία ζωτικής σημασίας σε κάθε εκπαιδευτικό σύστημα. Αυτή η πρακτική παρέχει στους αρμόδιους μηχανισμούς τα δεδομένα που απαιτούνται για να οργανώσουν και να διαμορφώσουν την εκπαίδευση των νεαρών κατοίκων της χώρας μας.

Παρόλα αυτά, το εκπαιδευτικό σύστημα της Ελλάδας παρουσιάζει αδυναμίες λόγω της εξαρτημένης σχέσης που έχει αναπτύξει με το παραδοσιακό απουσιολόγιο.

Για τον λόγο αυτό, παρατηρείται μια αυξημένη ανάγκη της ένταξης μιας νέας μεθόδου στη σύγχρονη ψηφιακή εποχή. Η απουσία κάποιας ευέλικτης και προσβάσιμης εφαρμογής για

την κάλυψη αυτής της ανάγκης είναι εμφανής. Η Students Absence καλείται να προσφέρει ακριβώς αυτή τη λύση, εστιάζοντας στην εύκολη και αποδοτική διαχείριση των απουσιών σε κάθε σχολείο με μια κοινή πλατφόρμα.

Παρακάτω θα δούμε ένα ορισμένο παράδειγμα εφαρμογής που προσφέρει τη δυνατότητα καταγραφής απουσιών:

Απουσιολόγιο

Το απουσιολόγιο είναι μια εφαρμογή με πολλαπλές λειτουργίες, σχεδιασμένη να λειτουργεί ως ένα εκτενές σύστημα e-class. Προσφέρει στους χρήστες τη δυνατότητα να προβάλλουν το εκπαιδευτικό πρόγραμμα, τους βαθμούς και άλλες βασικές σχολικές πληροφορίες, ενισχύοντας την επικοινωνία μεταξύ των μαθητών, των γονέων και των εκπαιδευτικών.

Στοχεύει στην άμεση ενημέρωση των μαθητών και των γονέων τους για σημαντικές πληροφορίες που τους αφορούν. Αυτό περιλαμβάνει την πρόσβαση σε εκπαιδευτικό υλικό, ανακοινώσεις, και άλλες σχετικές πληροφορίες, παρέχοντας ένα ολοκληρωμένο εργαλείο για τη διαχείριση της σχολικής ζωής.

Επιπρόσθετα, το Απουσιολόγιο περιλαμβάνει τη δυνατότητα προσωπικής καταχώρησης των απουσιών από τους ίδιους τους μαθητές. Αυτό επιτρέπει στους μαθητές να έχουν ενεργό ρόλο στην καταγραφή και παρακολούθηση των δικών τους απουσιών, ενθαρρύνοντας την αυτοδιαχείριση και την ευθύνη.

Ωστόσο, παρόλο που η εφαρμογή Απουσιολόγιο προσφέρει τη δυνατότητα στους μαθητές να καταχωρούν τις απουσίες τους, δεν καλύπτει τις λειτουργίες ενός παραδοσιακού απουσιολογίου.

Η ιδέα

Όπως αναφέρθηκε προηγουμένως, βρισκόμαστε σε μία εποχή ταχύτατης ψηφιοποίησης και καταχώρησης δεδομένων σε ηλεκτρονική μορφή. Επιπλέον, έχοντας ως δεδομένου το γεγονός πως η βιομηχανία χαρτιού καλείται να αντιμετωπίσει το πρόβλημα της έλλειψης λόγω της μείωσης του αριθμού των κομμένων δένδρων, χρειαζόμαστε την άμεση αντικατάσταση του φυσικού τετραδίου.

Με αφορμή τα μηχανήματα tablet που μοιράστηκαν σε σχολεία από την κυβέρνηση στη διάρκεια της τηλεκπαίδευσης, η καθηγήτρια πληροφορικής του 5ου Γενικού Λυκείου του Βύρωνα Ελένη Ρόμπολα και ο απόφοιτος μαθητής του σχολείου Ευστράτιος Δερμανούτσος αποφάσισαν να δώσουν τη λύση στο πρόβλημα.

Χρειαζόταν μια πλατφόρμα που αξιοποιεί τα tablet και, χωρίς κόπο ή τεχνολογικές γνώσεις, θα προσφέρει την πρόσβαση σε ένα πιο αποδοτικό απουσιολόγιο το οποίο είναι εύκολα διαχειρίσιμο από κάθε καθηγητή. Επίσης, επιλύει το πρόβλημα της μεταφοράς των δεδομένων στην κρατική πλατφόρμα του myschool που μέχρι σήμερα γίνεται χειροκίνητα.

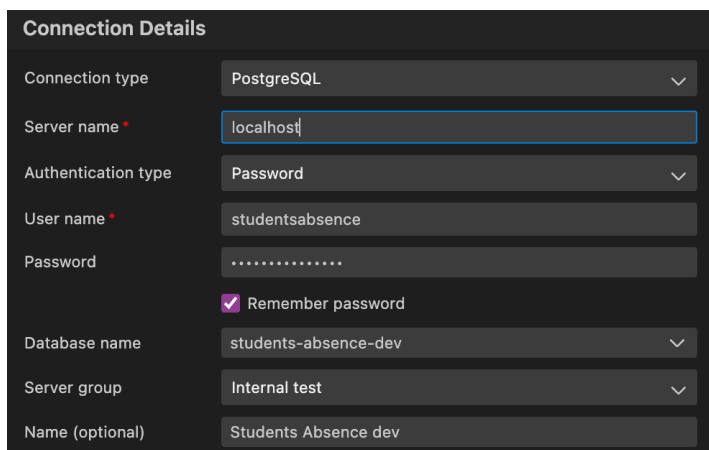
Εγκατάσταση της εφαρμογής

Βήμα 1: Εγκατάσταση της Βάσης Δεδομένων PostgreSQL

Για λόγους απλοποίησης αυτής της διαδικασίας, χρησιμοποιήθηκε έτοιμο image του προγράμματος Docker το οποίο θα εγκαταστήσουμε μέσω ενός αρχείου Docker Compose (**docker-compose.yml**).

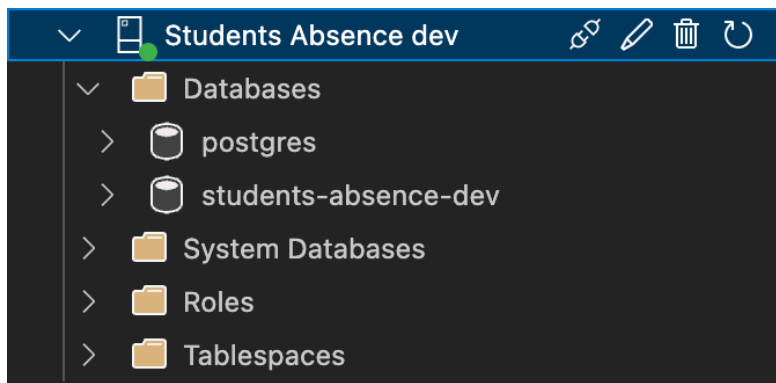
- Πρώτα απ' όλα, βεβαιωθείτε ότι έχετε εγκαταστήσει το Docker στο σύστημα σας.
- Ανοίξτε την εφαρμογή τερματικού (terminal) της προτίμησης σας και μεταβείτε στον φάκελο "docker" που θα βρείτε εντός του backend project.
- Εκτελέστε την παρακάτω εντολή:
`docker compose -f docker-compose.yml up -d`
- Για να επιβεβαιώσετε ότι η βάση δεδομένων είναι ενεργή και τρέχει, συνδεθείτε μέσω μιας εφαρμογής Διαχείρισης Βάσεων Δεδομένων.

Ακολουθούν screenshots από το Azure Data Studio της Microsoft.



The screenshot shows the 'Connection Details' window in Azure Data Studio. The configuration is as follows:

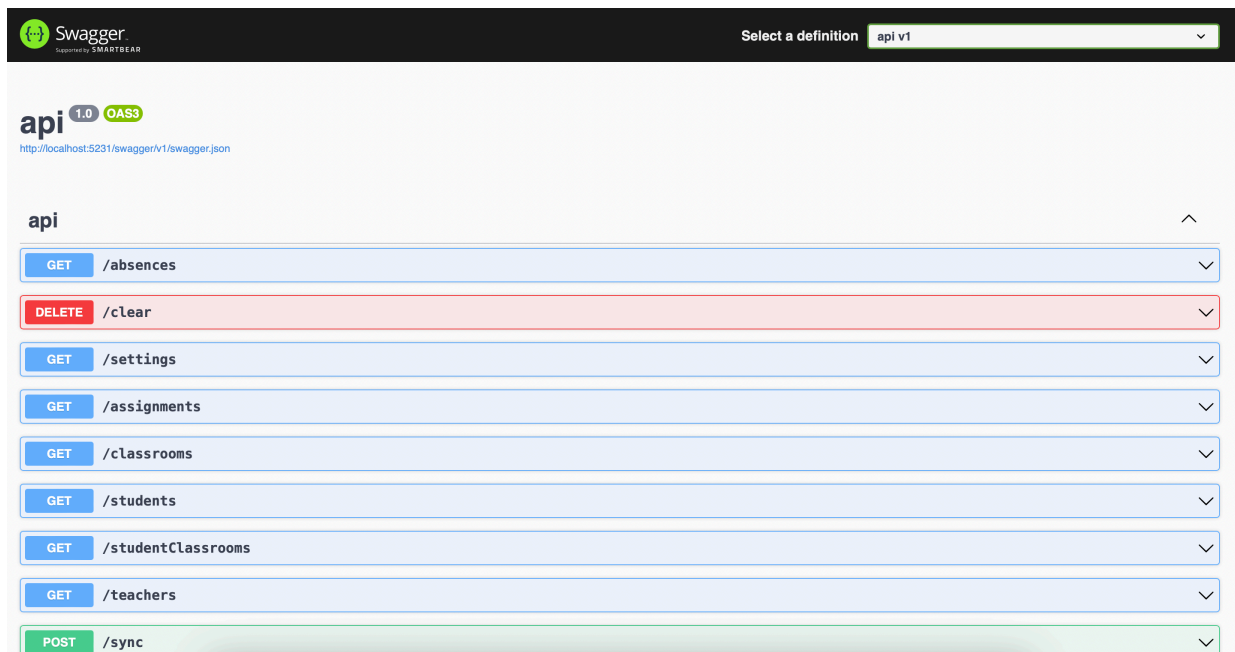
Field	Value
Connection type	PostgreSQL
Server name *	localhost
Authentication type	Password
User name *	studentsabsence
Password
<input checked="" type="checkbox"/> Remember password	
Database name	students-absence-dev
Server group	Internal test
Name (optional)	Students Absence dev



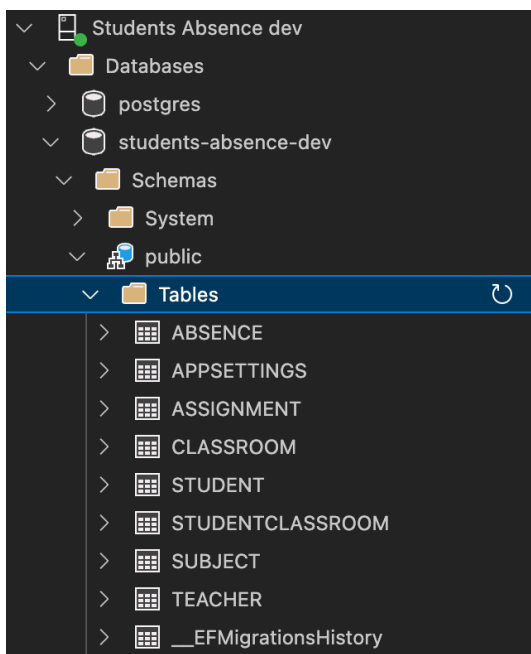
Σε αυτή την περίπτωση, ξέρουμε ότι η βάση λειτουργεί κανονικά και το Backend είναι έτοιμο να τη χρησιμοποιήσει.

Βήμα 2: Εγκατάσταση του Backend (.NET 8)

- Πρώτα απ' όλα, βεβαιωθείτε ότι έχετε εγκαταστήσει .NET 8 στο σύστημα σας.
- Εγκατάσταση του ASP.NET runtime.
Είναι απαραίτητο για να τρέξει το Web API.
- Για την εκκίνηση του server ανοίξτε το **api.sln** αρχείο μέσα από το Visual Studio της Microsoft, Visual Studio Code επίσης της Microsoft ή του Rider από την JetBrains.
Στην συνέχεια, πιάστε το κουμπί "Run".
- Αν η διαδικασία έγινε σωστά, θα ανοίξει ο φυλλομετρητής (browser) σας και θα οδηγηθείτε στην παρακάτω σελίδα.



Επιπλέον, θα επιβεβαιώσουμε ότι έτρεξε σωστά το API αν η βάση μας έχει λάβει τους πίνακες που χρησιμοποιούνται.



Να προσθέσω ότι αν δοκιμάσουμε να φέρουμε δεδομένα ενός πίνακα (π.χ. STUDENT) μέσω της SQL εντολής "SELECT" θα παρατηρήσουμε ότι υπάρχουν ήδη δεδομένα για λόγους δοκιμών.

Βήμα 3: Εγκατάσταση της εφαρμογής κινητού

Η εφαρμογή του κινητού έχει γραφεί με την χρήση της τεχνολογίας React Native ώστε να παίζει σε κάθε κινητή συσκευή.

Για την εκτέλεση της εφαρμογής από τον υπολογιστή μας για λόγους δοκιμών, θα πρέπει να έχουμε εγκατεστημένο κάποιο emulator.

- Πρώτα απ' όλα, βεβαιωθείτε ότι έχετε εγκαταστήσει το Node.js στο σύστημα σας.
Ειδανικά, θα εγκαταστήσετε την τελευταία Long-Term Support (LTS) έκδοση.
- Στη συνέχεια ανοίξτε την εφαρμογή τερματικού (terminal) της προτίμησής σας και μεταβείτε στον φάκελο του frontend project με την ονομασία "app-teacher".
- Τρέξτε την εντολή "npm install".

Βήμα 4: Προετοιμασία για την εκτέλεση της εφαρμογής

Για αυτό το βήμα, συνίσταται να ακολουθηθούν οι οδηγίες που θα βρείτε στο [official documentation της React Native](#).

Η ανάπτυξη της εφαρμογής έγινε από συσκευή Macbook της Apple με εικονική Android συσκευή.

Για τον λόγο αυτό, οι οδηγίες που ακολουθούν αναφέρονται σε αυτή την περίπτωση.

Συγκεκριμένα, θα χρειαστεί να έχουμε τα παρακάτω προαπαιτούμενα:

- Node.js (το εγκαταστήσαμε σε προηγούμενο βήμα)
- Watchman
- React Native command line interface
Το συγκεκριμένο θα εγκατασταθεί από μόνο του τοπικά στο project μας.
- JDK (στην περίπτωση μας έχουμε το 17)
Εκτελεστέ την εντολή "brew install openjdk@17" σε ένα παράθυρο τερματικού.
- Android Studio

Πιο αναλυτικά, θα ξεκινήσουμε εγκαθιστώντας τα Node και Watchman μέσω του brew package manager για εφαρμογές τερματικού που παρέχει το MacOS

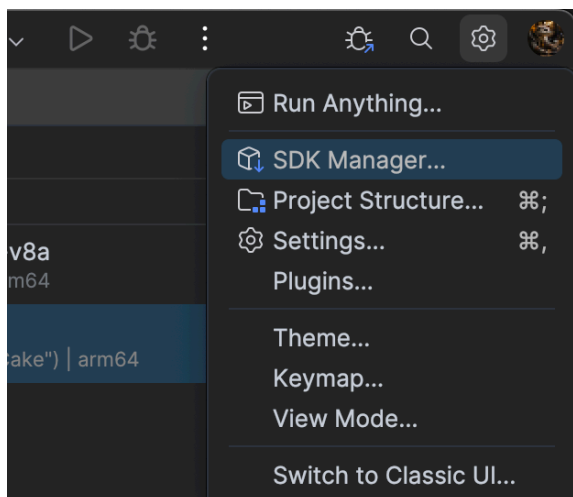
- Εκτελέστε "brew install node" σε ένα τερματικό
- Εκτελέστε "brew install watchman" σε ένα τερματικό
- Εκτελέστε "brew install openjdk@17" σε ένα τερματικό
Για να βεβαιωθείτε ότι εγκαταστήθηκε σωστά, εκτελέστε στο ίδιο τερματικό "java --version"
Το αποτέλεσμα θα πρέπει να είναι παρόμοιο με το παρακάτω.

```
stratis@192 app-teacher % java --version
openjdk 17.0.12 2024-07-16
OpenJDK Runtime Environment Homebrew (build 17.0.12+0)
OpenJDK 64-Bit Server VM Homebrew (build 17.0.12+0, mixed mode, sharing)
```

- Εγκατάσταση του Android Studio.
Για πιο εύκολη εγκατάσταση και μελλοντική αναβάθμιση, συνιστάται η χρήση του JetBrains Toolbox.
- Εγκατάσταση του Android SDK.
Το Android Studio εγκαθιστά το πιο πρόσφατο Android SDK από προεπιλογή.

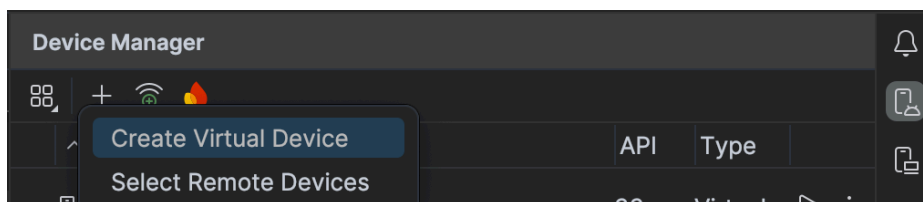
Για την εφαρμογή μας, χρησιμοποιήθηκε η έκδοση “Android 14 (UpsideDownCake)” που μπορείτε να το βρείτε στο SDK Manager.

Ενδεικτικά:



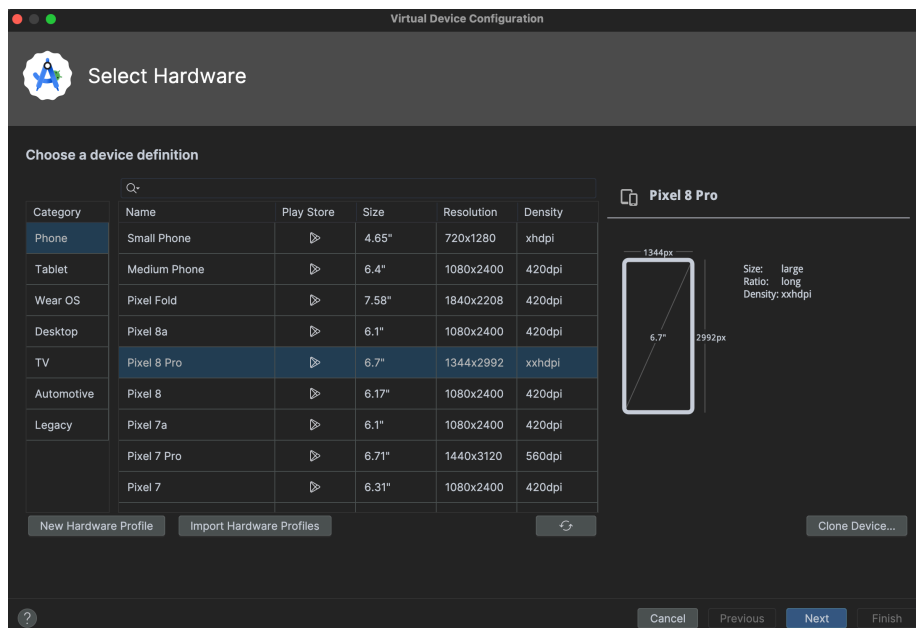
	Name	API Level	Re...	Status
<input type="checkbox"/>	Android 15.0 ("VanillalceCream")	35	1	Not installed
<input type="checkbox"/>	Android VanillalceCream Preview	VanillalceCream	4	Not installed
<input type="checkbox"/>	Android UpsideDownCakePrivacySandbox Prev UpsideDownCakePrivacySan...	UpsideDownCakePrivacySan...	3	Not installed
<input checked="" type="checkbox"/>	Android 14.0 ("UpsideDownCake")	34	3	Installed

- Δημιουργήστε εικονική συσκευή Android ακολουθώντας τα βήματα που απεικονίζονται παρακάτω:



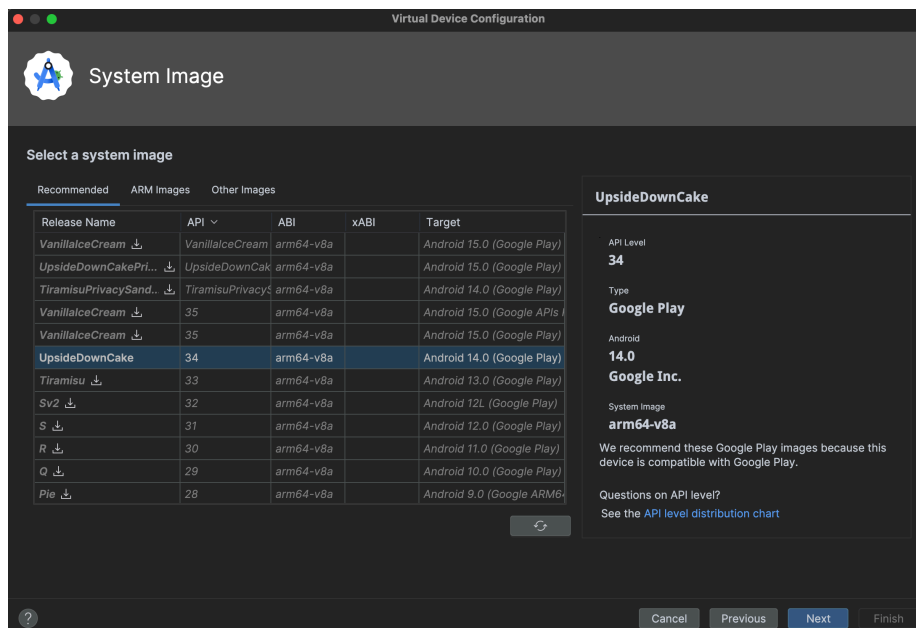
1. Επιλογή συσκευής

Επιλέξαμε το Pixel 8 Pro για την μεγάλη οθόνη αλλά δεν έχει σημασία.

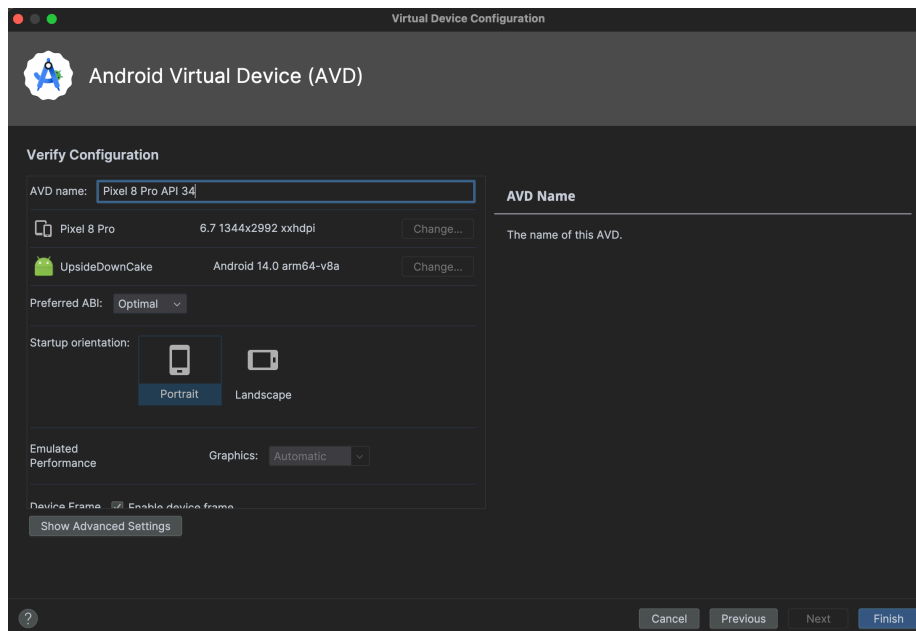


2. Εικόνα συστήματος

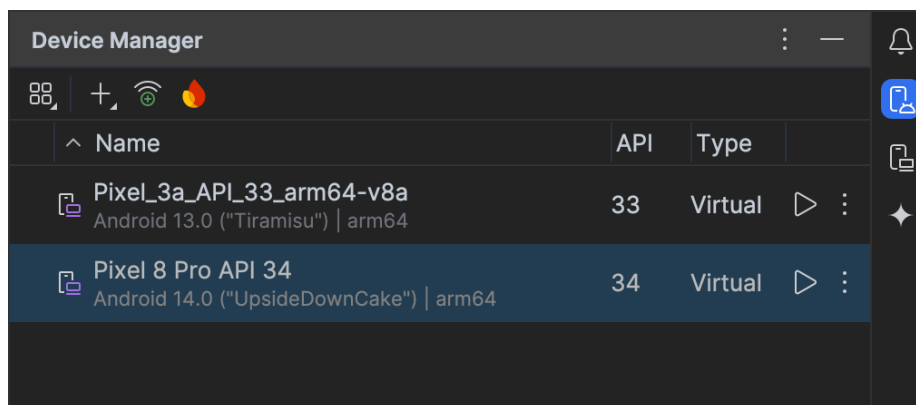
Εμείς διαλέξαμε την έκδοση 34 που αντιστοιχεί στο SDK μας.



3. Λοιπές ρυθμίσεις για την εικονική συσκευή Android / Android Virtual Device (AVD)

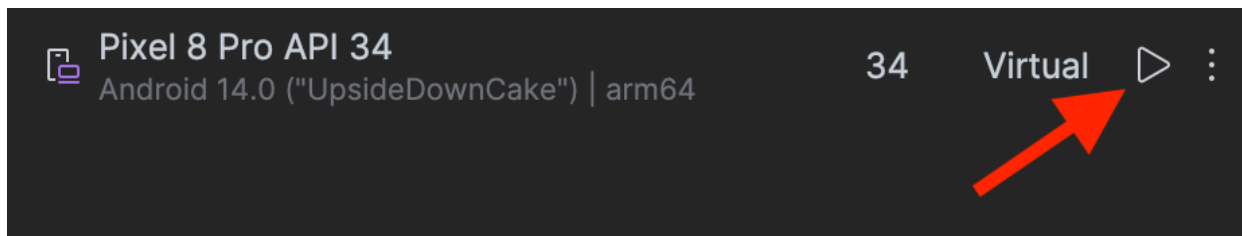


4. Τέλος, η συσκευή σας θα πρέπει να φαίνεται στο μενού "Device Manager"



Βήμα 5: Εκτέλεση της εφαρμογής

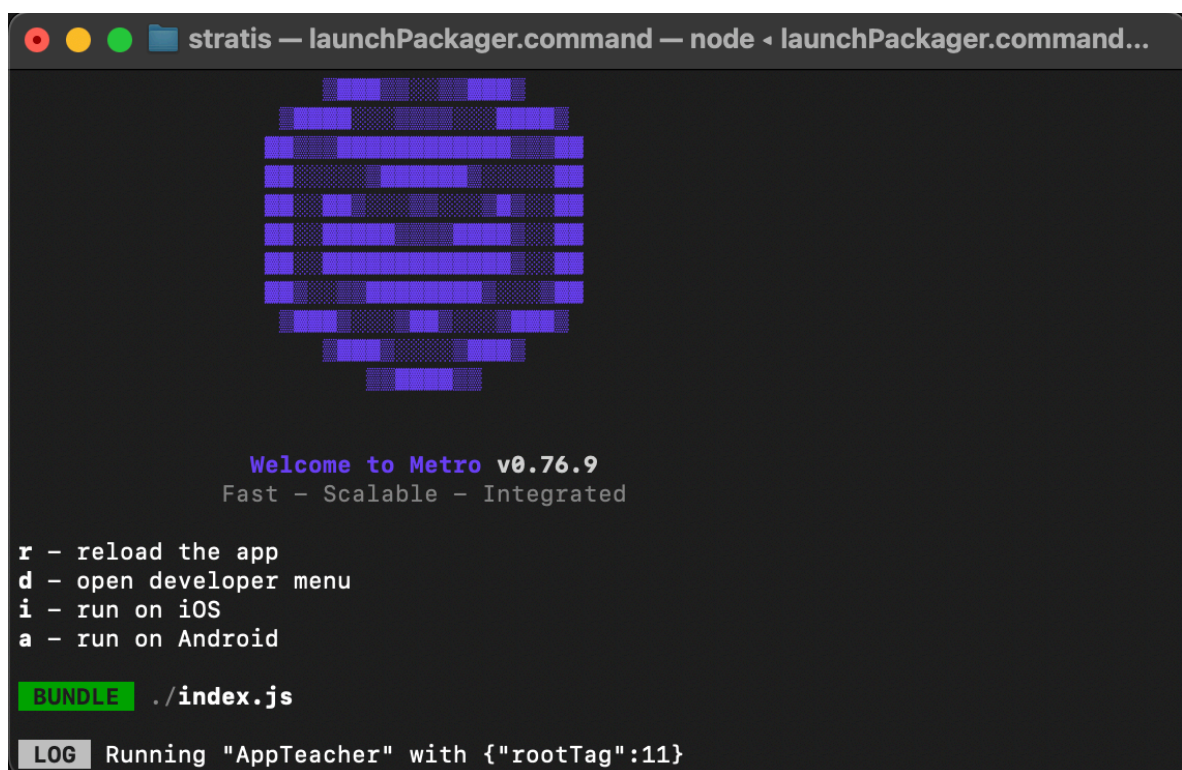
- Ξεκινήστε την εικονική συσκευή που δημιουργήσατε στο προηγούμενο βήμα πατώντας το κουμπί "Run" δίπλα από την περιγραφή της συσκευής.



- Αφού η συσκευή ξεκινήσει, θα τη δείτε στο παράθυρο του Android Studio όπως παρακάτω.



- Στη συνέχεια ανοίξτε την εφαρμογή τερματικού (terminal) της προτίμησής σας και μεταβείτε στον φάκελο του frontend project με την ονομασία "app-teacher".
- Τρέξτε την εντολή "npm run android".
Μετά από λίγη ώρα που χρειάζεται η React Native για να παράξει το APK αρχείο και να το εγκαταστήσει στο εικονικό κινητό μας τηλέφωνο, θα δούμε δύο πράγματα:
 1. Ένα τερματικό να τρέχει με την εφαρμογή "Metro"



The screenshot shows a terminal window titled "stratis — launchPackager.command — node < launchPackager.command...". The terminal displays the Metro bundler logo, a welcome message "Welcome to Metro v0.76.9" with the tagline "Fast - Scalable - Integrated", and a list of commands: "r - reload the app", "d - open developer menu", "i - run on iOS", and "a - run on Android". Below this, a green bar indicates "BUNDLE ./index.js" and a log message shows "LOG Running 'AppTeacher' with {'rootTag':11}".

```
stratis — launchPackager.command — node < launchPackager.command...

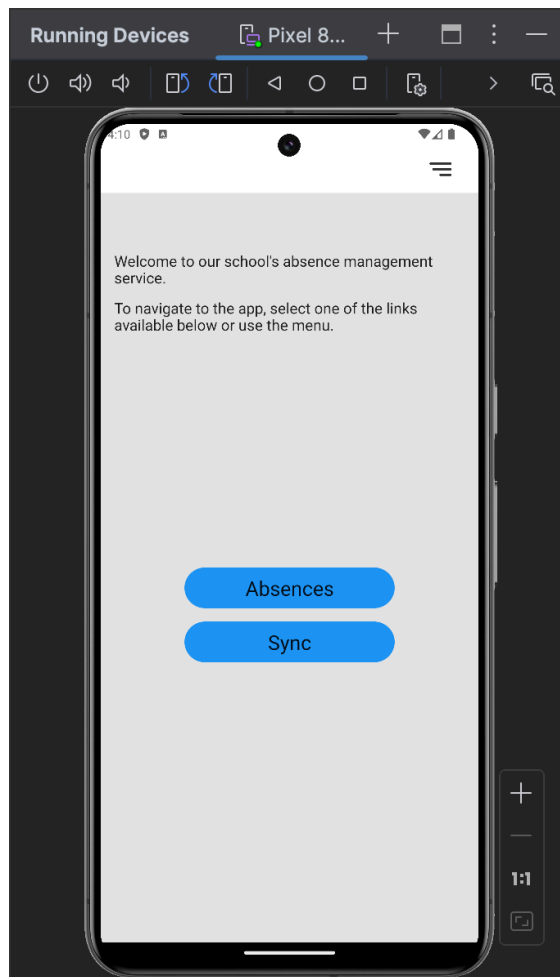
Welcome to Metro v0.76.9
Fast - Scalable - Integrated

r - reload the app
d - open developer menu
i - run on iOS
a - run on Android

BUNDLE ./index.js

LOG Running "AppTeacher" with {"rootTag":11}
```

2. Η οθόνη της εικονικής κινητής συσκευής να δείχνει την εφαρμογή.



Εγχειρίδιο χρήστη / User manual

Όταν ανοίξετε την εφαρμογή, θα μεταβείτε στην αρχική σελίδα "Home", η οποία λειτουργεί και ως βασικό μενού για την περιήγηση σε όλες τις σελίδες της εφαρμογής.

Την οθόνη αυτή θα την δείτε στην παραπάνω εικόνα.

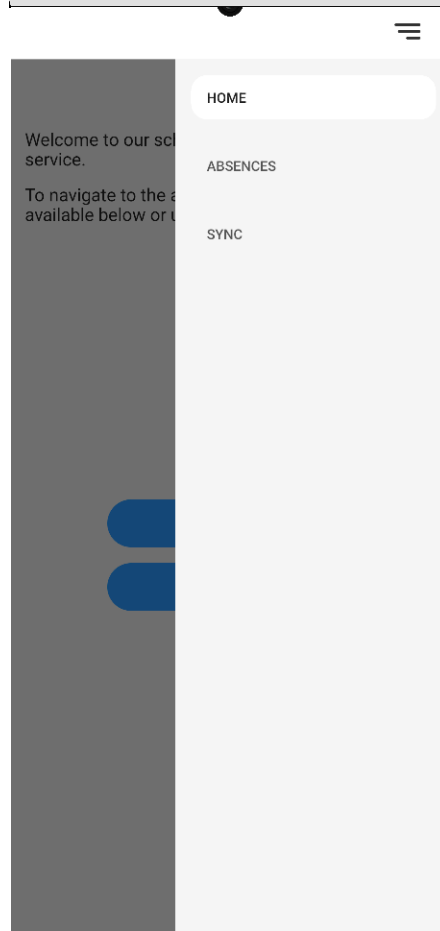
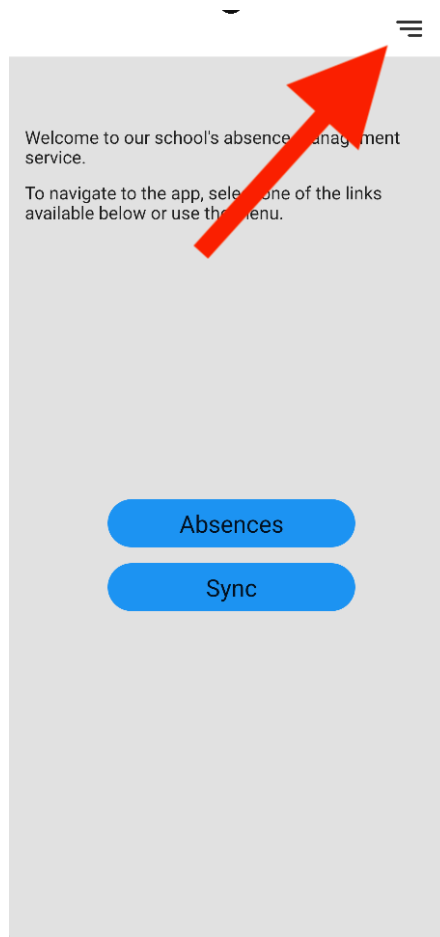
Εισαγωγική περιγραφή

Η διεπαφή χρήστη με την οποία ερχόμαστε σε επαφή έχει δημιουργηθεί με βάση τα πρότυπα του Material Design και στοχεύει στη δημιουργία μιας απλής, φιλικής και συνεκτικής εμπειρίας για τον χρήστη. Τα οπτικά στοιχεία είναι διακριτικά και καλά οργανωμένα, ενώ τα χρώματα και οι διατάξεις έχουν επιλεγεί προσεκτικά για να διευκολύνουν την αλληλεπίδραση και την ανάγνωση πληροφοριών. Η πλοήγηση στο σύστημα είναι εύκολη και προσιτή σε όλες τις κατηγορίες χρηστών, δίνοντας έμφαση στην ευκολία χρήσης χωρίς να απαιτούνται τεχνικές γνώσεις.

Κεντρικό μενού

Για την πλοήγηση του χρήστη σε οποιαδήποτε οθόνη της εφαρμογής, έχει δημιουργηθεί ένα κεντρικό μενού με συνδέσμους που προσφέρουν άμεση πρόσβαση στις κύριες λειτουργίες της. Οι επιλογές είναι οργανωμένες με τρόπο που επιτρέπει στον χρήστη να βρίσκει εύκολα αυτό που χρειάζεται, ενώ το μενού παραμένει διαθέσιμο σε όλες τις οθόνες για γρήγορη και αποτελεσματική πλοήγηση. Το μενού αυτό έχει σχεδιαστεί με γνώμονα την απλότητα και τη σαφήνεια, ώστε να μειώνεται ο χρόνος που απαιτείται για την εύρεση των βασικών λειτουργιών.

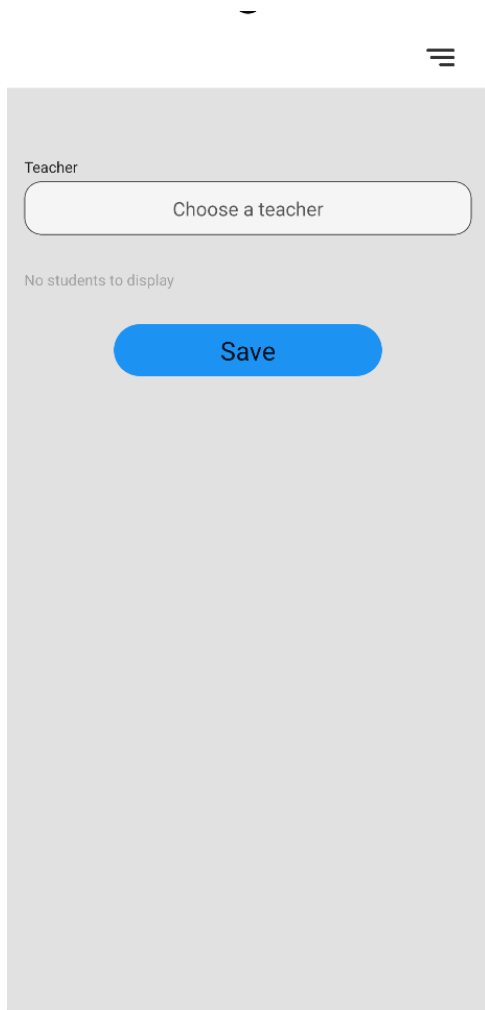
Για την εμφάνιση του μενού, αρκεί να επιλέξουμε το κουμπί “Hamburger” στην πάνω δεξιά γωνία της οθόνης και θα “συρθεί” το μενού στο οπτικό μας πεδίο.



Επιλογές χρήστη

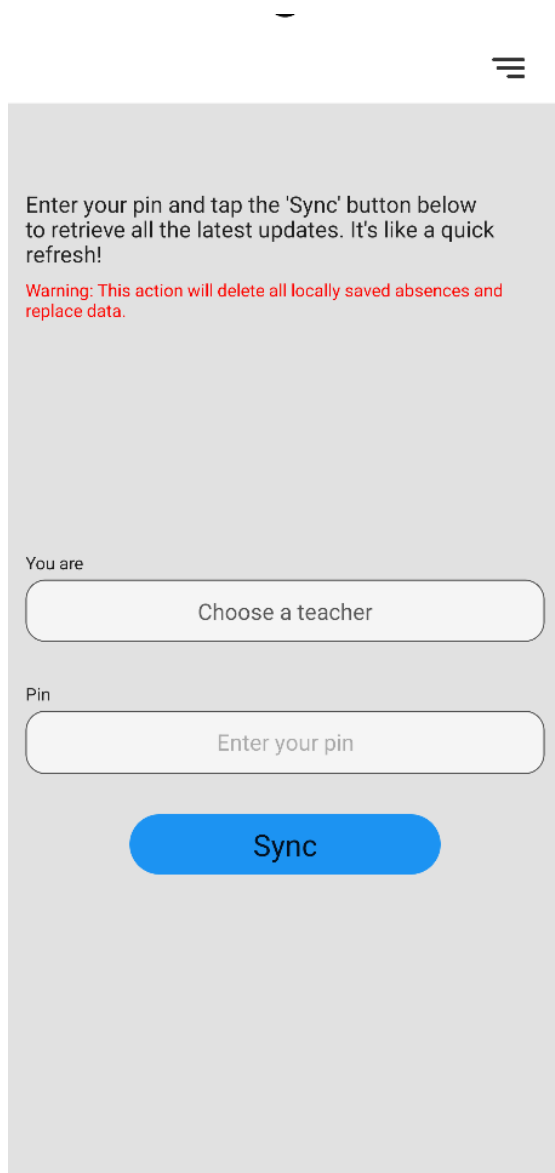
Ο χρήστης έχει τη δυνατότητα να επιλέξει ανάμεσα στις εξής λειτουργίες:

- Αν επιλέξετε το **“Home”** / **“Αρχική”** θα μεταβείτε στην αρχική σελίδα menu όπως βλέπετε παραπάνω.
- Εάν επιλέξετε **“Absences”** / **“Απουσίες”** θα μεταβείτε στο τμήμα που παρουσιάζονται και επεξεργάζονται οι απουσίες των μαθητών ανά διδακτική ώρα.



The screenshot shows a mobile application interface. At the top right, there is a hamburger menu icon. Below it, the word "Teacher" is displayed above a rounded rectangular input field containing the text "Choose a teacher". Underneath this field, the text "No students to display" is shown. At the bottom of the visible area, there is a prominent blue rounded rectangular button with the word "Save" in white text.

- Τέλος, η επιλογή **“Sync”** / **“Συγχρονισμός”** μας οδηγεί στην οθόνη συγχρονισμού όπου μοναδικός σκοπός της αποτελεί ο συγχρονισμός της συσκευής μας με τα τελευταία δεδομένα του server.



Πρώτη εκκίνηση

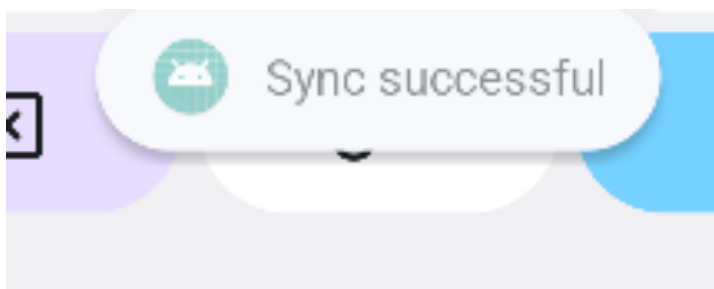
Όπως αναφέρθηκε προηγουμένως, η εφαρμογή στοχεύει σε καθηγητές σχολείων και έχει σκοπό τη βελτίωση των συνθηκών καταγραφής απουσιών.

Για το λόγο αυτό, η εφαρμογή έρχεται έτοιμη να καλύψει τις ανάγκες κάθε σχολείου στο οποίο θα χρησιμοποιηθεί. Επομένως, πρέπει το σχολείο να έχει μια ομάδα που θα διαχειρίζεται τα δεδομένα του σχολείου και, ακολούθως, της εφαρμογής.

Για να ξεκινήσουμε να τη χρησιμοποιούμε, πρέπει ένας διαχειριστής να μεταβεί στη σελίδα συγχρονισμού και, αφού πληκτρολογήσει το μυστικό του pin, με κενό το πεδίο του καθηγητή, να πιάσει το κουμπί που αναγράφει “Sync” / “Συγχρονισμός”.

Αν η επικοινωνία με τον διακομιστή έγινε σωστά, τα αποτελέσματα θα είναι τα εξής:

- Καθαρισμός των δεδομένων στην τοπική βάση SQLite του κινητού, καθώς πλέον οι απουσίες υπάρχουν στο κεντρικό σύστημα διαχείρισης.
- Λήψη νέων δεδομένων σε περίπτωση αλλαγών αυτών του συστήματος.
Π.χ. Προστέθηκε ένας νέος καθηγητής ή άλλαξαν οι ώρες διδασκαλίας κάποιου καθηγητή σε ένα τμήμα.
- Ειδοποίηση του χρήστη για την επιτυχή ολοκλήρωση της διαδικασίας συγχρονισμού με τη χρήση ειδοποίησης τύπου “Toast”.



Για λόγους test, ο κωδικός του διαχειριστή βρίσκεται hard-coded στον κώδικα και έχει την τιμή “1928”.

Μετά τον συγχρονισμό, θα παρατηρήσουμε ότι υπάρχουν και άλλες αλλαγές στη διεπαφή χρήστη της εφαρμογής.

Για παράδειγμα, υπάρχουν πλέον το λογότυπο (αν υπάρχει) και το όνομα του εκάστοτε σχολείου στην πάνω μπάρα.

Πριν



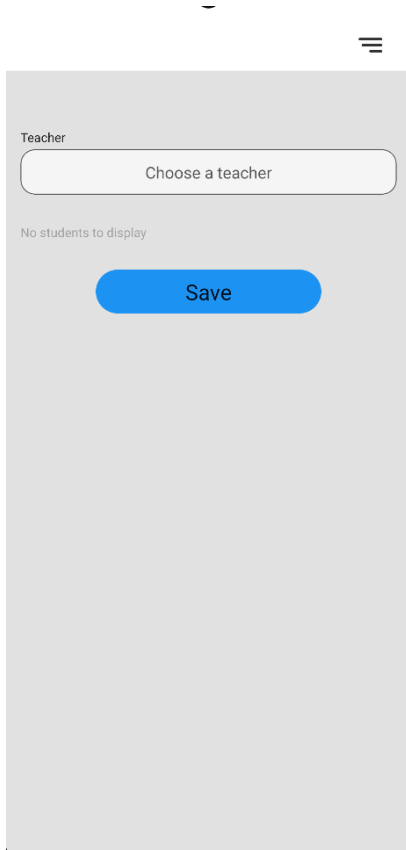
Μετά



Χρήση από το κοινό

Όπως αναφέρθηκε προηγουμένως, η εφαρμογή αυτή στοχεύει στη διευκόλυνση των καθηγητών ενός σχολείου. Σημαντικό στοιχείο για την επίτευξη αυτού του σκοπού είναι η έλλειψη περιττών κινήσεων και λειτουργιών που θα έμπλεκαν τη χρήση από όσους δεν είναι πλήρως εξοικειωμένοι με την τεχνολογία και τις φορητές συσκευές.

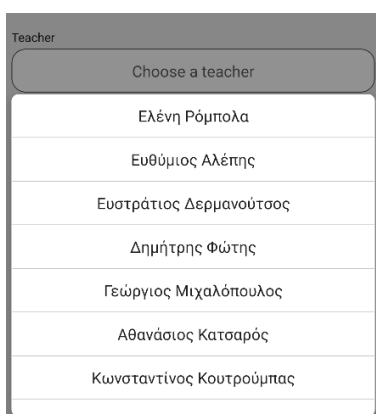
Για το λόγο αυτό, έναν καθηγητή που χρησιμοποιεί αυτή την εφαρμογή τον απασχολεί μόνο με μία βασική οθόνη, αυτή της καταχώρισης απουσιών.



The screenshot shows a mobile app interface for selecting a teacher. At the top, there is a hamburger menu icon. Below it, the word "Teacher" is displayed. A text input field with the placeholder "Choose a teacher" is present. Below the input field, the text "No students to display" is shown. At the bottom, there is a blue button labeled "Save".

Στην παραπάνω οθόνη, υπάρχει ένα πεδίο επιλογής καθηγητή και ένα μήνυμα ενημέρωσης πως δεν υπάρχουν, ακόμα, μαθητές προς εμφάνιση.

Για να συνεχίσουμε, πρέπει να επιλέξουμε τον καθηγητή ενδιαφέροντος μέσω του πεδίου.



The screenshot shows the same mobile app interface as before, but with a list of teachers displayed below the input field. The list contains the following names: Ελένη Ρόμπολα, Ευθύμιος Αλέπης, Ευστράτιος Δερμανούτσος, Δημήτρης Φώτης, Γεώργιος Μιχαλόπουλος, Αθανάσιος Κατσάρος, and Κωνσταντίνος Κουτρούμπας.

Τα ονόματα που φαίνονται στη λίστα είναι ενδεικτικά και απεικονίζουν απαραίτητα πραγματικά δεδομένα.

Έχοντας, λοιπόν, επιλέξει τον καθηγητή, θα μας εμφανιστεί άλλο ένα πεδίο που μας επιτρέπει να επιλέξουμε και την ανάθεση του για εκείνη την διδακτική ώρα.

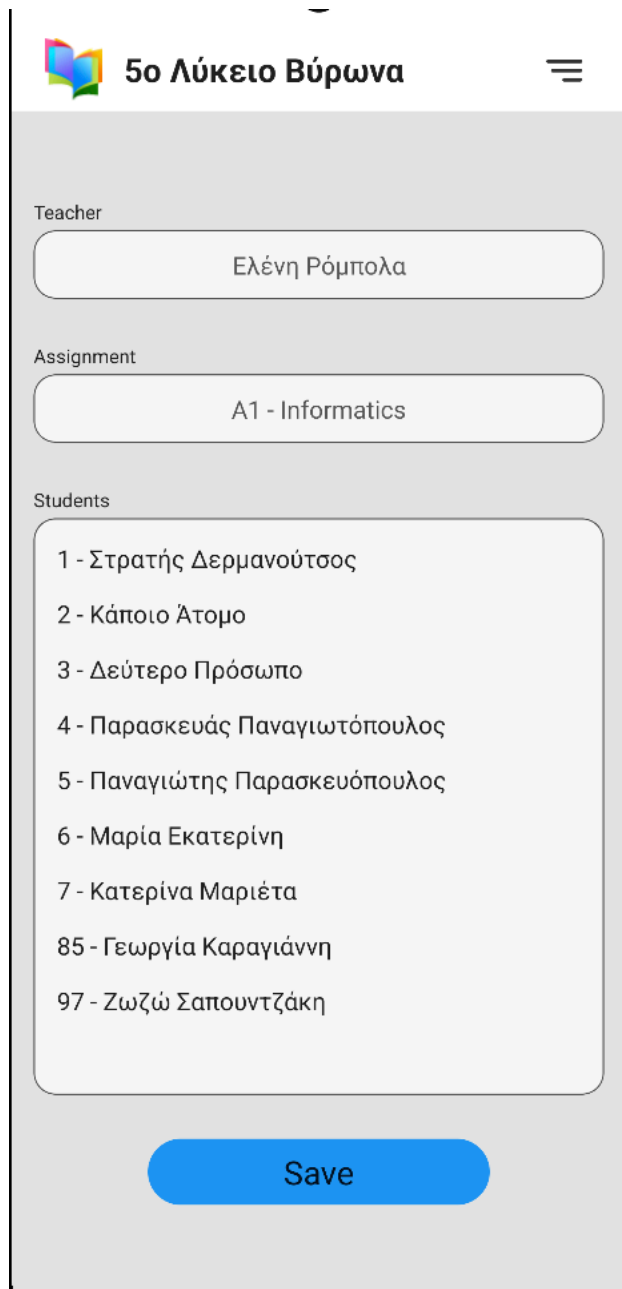
Η ανάθεση ενός καθηγητή είναι συνδυασμός του τμήματος και του μαθήματος που διδάσκει. Παρακάτω φαίνονται οι πιθανές επιλογές για την καθηγήτρια “Ελένη Ρόμπολα”:

Teacher
Ελένη Ρόμπολα
Assignment
Choose an assignment
A1 - Informatics
A2 - Mathematics
B1 - Physics

Όπως είναι λογικό, οι επιλογές ανάθεσης διαφέρουν από καθηγητή σε καθηγητή.

Teacher
Ευστράτιος Δερμανούτσος
Assignment
Choose an assignment
B1 - Chemistry
B2 - Political Economy
B3 - Political Economy

Μετά την επιλογή κάποιας ανάθεσης, εμφανίζεται η λίστα όλων των μαθητών που αντιστοιχούν στο τμήμα.



The screenshot shows a mobile application interface for the '5ο Λύκειο Βύρωνα' (5th Lyceum of Vryon). The interface is divided into three main sections: Teacher, Assignment, and Students. The Teacher section shows 'Ελένη Ρόμπολα'. The Assignment section shows 'A1 - Informatics'. The Students section lists nine students, with the last two (85 and 97) highlighted in red. A blue 'Save' button is at the bottom.

5ο Λύκειο Βύρωνα

Teacher

Ελένη Ρόμπολα

Assignment

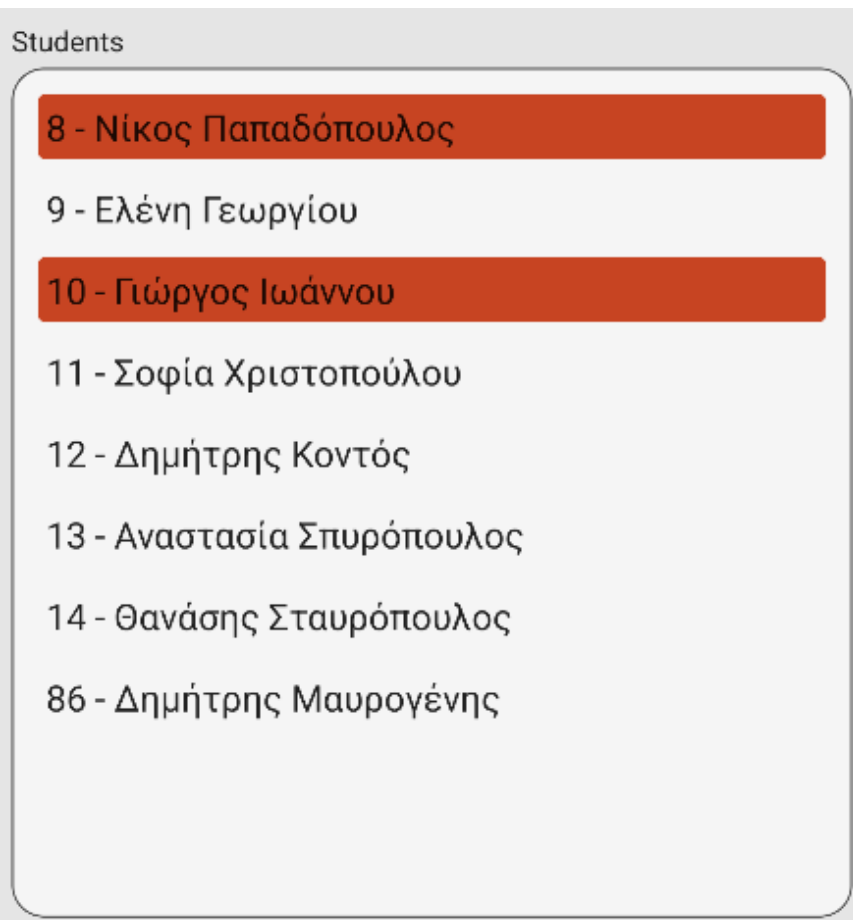
A1 - Informatics

Students

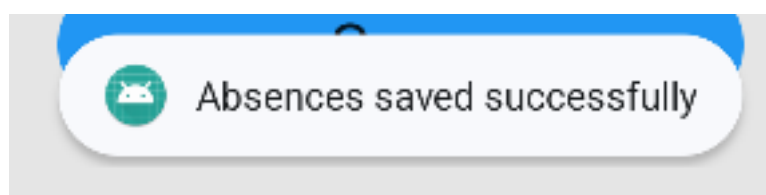
- 1 - Στρατής Δερμανούτσος
- 2 - Κάποιο Άτομο
- 3 - Δεύτερο Πρόσωπο
- 4 - Παρασκευάς Παναγιωτόπουλος
- 5 - Παναγιώτης Παρασκευόπουλος
- 6 - Μαρία Εκατερίνη
- 7 - Κατερίνα Μαριέτα
- 85 - Γεωργία Καραγιάννη
- 97 - Ζωζώ Σαπουντζάκη

Save

Η εφαρμογή, επίσης, θυμάται για την ίδια μέρα και ίδια ανάθεση ποιοι μαθητές έλειπαν από την αίθουσα του μαθήματος και τους σημειώνει με κόκκινο πλαίσιο.



Μετά την επιλογή του κουμπιού “Save” / “Αποθήκευση”, μας εμφανίζεται το αντίστοιχο Toast μήνυμα.



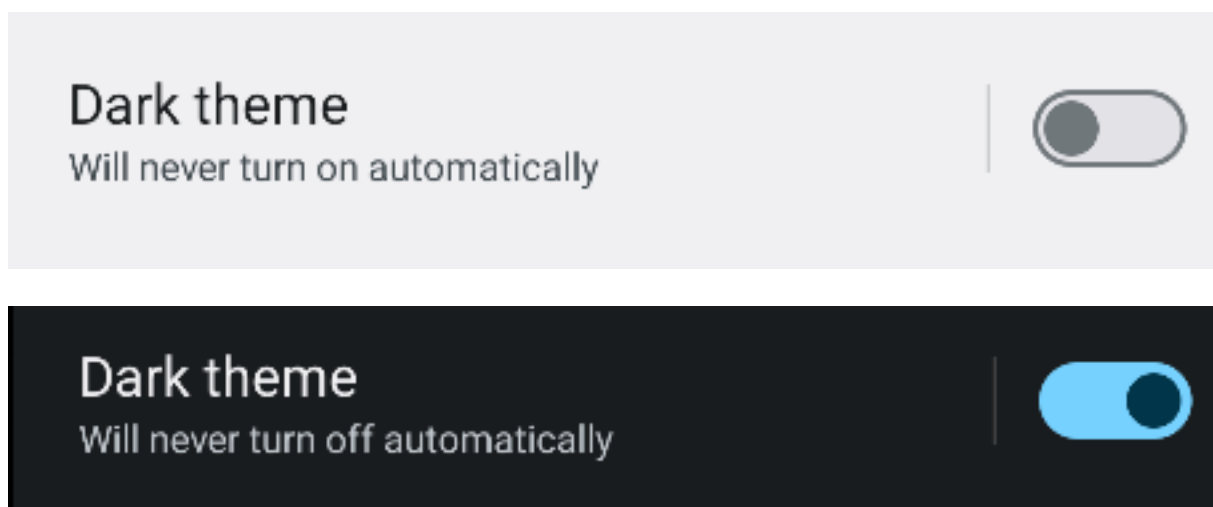
Η επεξεργασία των απουσιών είναι εφικτή μόνο την ίδια ημέρα για λόγους κάλυψης πιθανών λαθών.

Επιπλέον χαρακτηριστικά

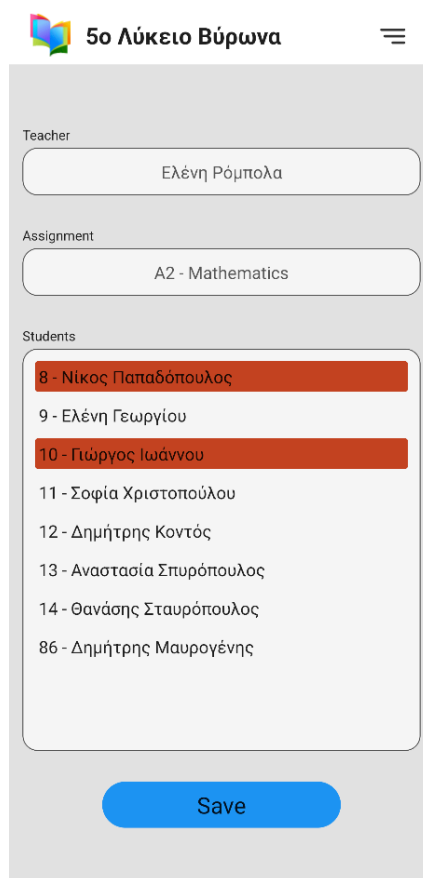
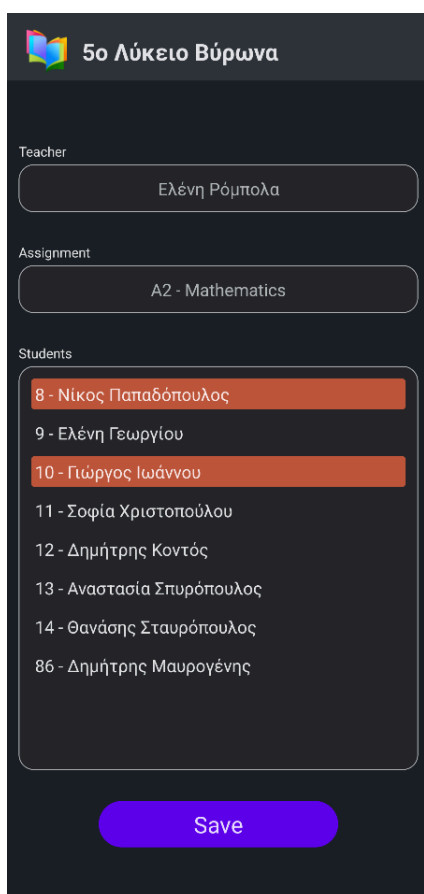
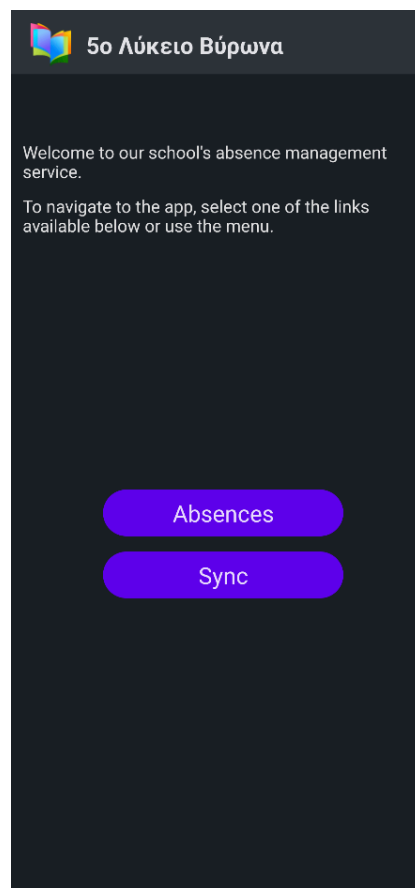
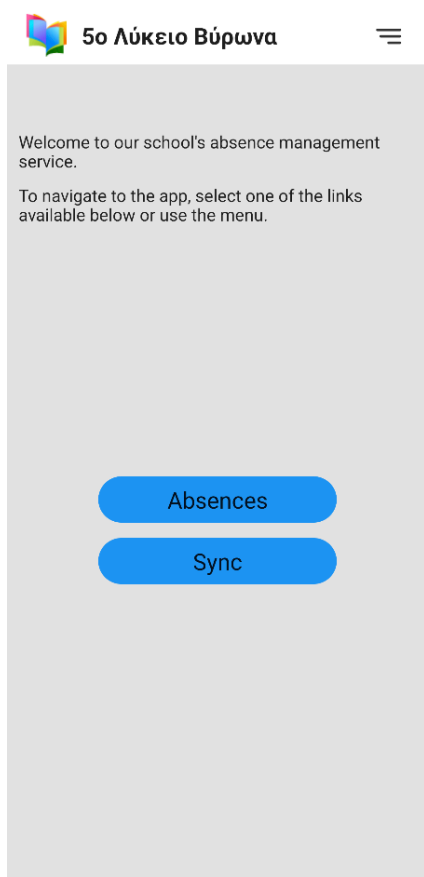
Σκοτεινό θέμα

Η εφαρμογή προσφέρει τη δυνατότητα εναλλαγής μεταξύ φωτεινού και σκοτεινού θέματος για να προσαρμόσετε την εμφάνιση της σύμφωνα με τις προτιμήσεις σας ή τις συνθήκες φωτισμού. Το σκοτεινό θέμα είναι ιδιαίτερα χρήσιμο σε περιβάλλοντα με χαμηλό φωτισμό, καθώς μειώνει την ένταση του φωτός της οθόνης και μπορεί να είναι πιο ξεκούραστο για τα μάτια.

Ενεργοποίηση Σκοτεινού Θέματος:



Παρακάτω θα δείτε σύγκριση των διάφορων οθονών με τα δύο θέματα.

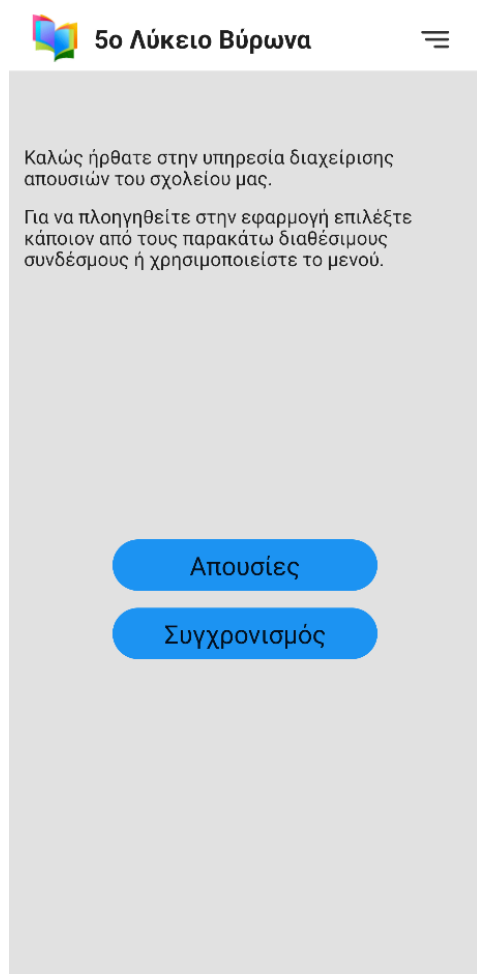


Υποστήριξη Γλωσσών (Localization)

Η εφαρμογή υποστηρίζει πολλαπλές γλώσσες, επιτρέποντας σας να την χρησιμοποιείτε στη γλώσσα που σας εξυπηρετεί καλύτερα. Μπορείτε να αλλάξετε τη γλώσσα της διεπαφής εύκολα μέσω των ρυθμίσεων της εφαρμογής.

Με βάση τα σχέδια επέκτασης σε σχολεία τους εξωτερικού, η υποστήριξη πολλαπλών γλωσσών κρίθηκε απαραίτητο χαρακτηριστικό της εφαρμογής.

Αν αλλάξουμε τη γλώσσα του συστήματος στα ελληνικά και επιστρέψουμε στην εφαρμογή, θα δούμε ότι πλέον όλα τα λεκτικά είναι γραμμένα στην ελληνική γλώσσα.



Αρχιτεκτονική συστήματος

Η εφαρμογή Students Absence εφαρμόζει προηγμένες τεχνικές για να εξασφαλίσει ομαλή και γρήγορη καταγραφή απουσιών σε κάθε συσκευή, χωρίς να επηρεάζεται η απόδοση. Με μια συνδυασμένη προσέγγιση στην τεχνολογία, η Students Absence προσφέρει άμεση πρόσβαση στα δεδομένα των μαθητών, ενώ ταυτόχρονα ελαχιστοποιεί τη χρήση δεδομένων, βελτιώνοντας την εμπειρία του χρήστη και την αποδοτικότητα στην καταγραφή των απουσιών.

Βάση Δεδομένων: **PostgreSQL**

Backend: **.NET 8**

Frontend : **Javascript**

Framework: **React Native**

Παρουσίαση βάσης δεδομένων

Η βάση δεδομένων PostgreSQL ονομάζεται students-absence-dev και αποτελείται από 9 tables:

- **ABSENCE**

Ο βασικός πίνακας αποθήκευσης των απουσιών.

- **APPSETTING**

Ο πίνακας που κρατάει τις πληροφορίες για την εφαρμογή. Συγκεκριμένα, περιέχεται το όνομα και ο σύνδεσμος για την εικόνα του εκάστοτε σχολείου.

- **ASSIGNMENT**

Πίνακας υπεύθυνος για την αντιστοίχιση καθηγητών και μαθημάτων με κάθε τάξη.

Κάθε εγγραφή έχει μοναδικό Id καθώς και τα Id των 3 tables που συσχετίζει. (CLASSROOM, SUBJECT, TEACHER)

- **CLASSROOM**

Οι ονομασίες των τάξεων στην αγγλική, καθώς και την εγγενή, γλώσσα και το μοναδικό Id.

- **STUDENT**

Οι πληροφορίες του κάθε μαθητή του σχολείου.

- **STUDENTCLASSROOM**

Πίνακας αντιστοίχισης μαθητών και τάξεων..

Κάθε εγγραφή έχει μοναδικό τα Id των 2 tables που συσχετίζει. (CLASSROOM, STUDENT)

- **SUBJECT**

Οι ονομασίες των μαθημάτων στην αγγλική, καθώς και την εγγενή, γλώσσα και το μοναδικό Id.

- **TEACHER**

Οι πληροφορίες των καθηγητών του σχολείου καθώς και το μοναδικό pin του καθενός σε κρυπτογραφημένη μορφή.

- **__EFMigrationHistory**

Ιστορικό εκδόσεων της βάσης. Ο συγκεκριμένος πίνακας δημιουργείται και συμπληρώνεται αυτόματα από την βιβλιοθήκη Entity Framework.

Παρουσίαση backend

Για τη δημιουργία του backend, χρησιμοποιήθηκε το πρότυπο “Minimal APIs” που προσφέρει η Microsoft για Web API project από την .NET 6 και μετά.

Endpoints

- **/absences [GET]**
Επιστρέφει όλες τις απουσίες που έχουν περαστεί στο σύστημα.
- **/assignments [GET]**
Επιστρέφει τις αναθέσεις των καθηγητών σε κάθε τάξη.
- **/classrooms [GET]**
Επιστρέφει τις πληροφορίες των τάξεων
- **/clear [DELETE]**
Καθαρίζει όλες τις απουσίες. Η χρήση του είναι όταν έχουν περάσει πλέον οι απουσίες στην πλατφόρμα MySchool, να καθαρίσει το εσωτερικό σύστημα του σχολείου από τα παραπάνω δεδομένα.
- **/settings [GET]**
Επιστρέφει τις ρυθμίσεις του εκάστοτε σχολείου. Στην τρέχουσα έκδοση της εφαρμογής, επιστρέφει το όνομα και το λογότυπο.
- **/students [GET]**
Επιστρέφει όλους τους μαθητές του σχολείου.
- **/studentClassrooms [GET]**
Επιστρέφει όλες τις συσχετίσεις μαθητών και τάξεων.

- **/sync [POST]**
Επιβεβαιώνει το μοναδικό κωδικό του καθηγητή που το χρησιμοποίησε και προχωράει στον συγχρονισμό της συσκευής από την οποία κλήθηκε.
- **/teachers [GET]**
Επιστρέφει όλους τους καθηγητές που εργάζονται στο σχολείο.

Παρουσίαση frontend

Το frontend μέρος της εφαρμογής έχει φτιαχτεί με τη χρήση του framework “React Native” που μας επιτρέπει να γράφουμε React εφαρμογές και τις μεταφράζει σε native κώδικα για Android και iOS αντίστοιχα.

Δομή φακέλων

Η εφαρμογή μας έχει φτιαχτεί από τρεις (3) βασικούς φακέλους που περιέχουν τον κώδικα μας:

- **components/**: Περιέχει όλα τα αντικείμενα που αποτελούν την διεπαφή με τον χρήστη (UI). Τέτοια αντικείμενα μπορεί να είναι οθόνες, κουμπιά, μενού κλπ.
- **data/**: Περιέχει οτιδήποτε μπορεί να θεωρηθεί δεδομένο ή να χρησιμοποιηθεί για την διαχείριση των δεδομένων. Αυτό συμπεριλαμβάνει βοηθητικές μεθόδους και κλάσεις, enumerations, μοντέλα και custom hooks της React.
- **resources/**: Σε αυτό τον φάκελο υπάρχουν όλα τα άλλα αρχεία που χρησιμοποιήθηκαν για τη δημιουργία της εφαρμογής.

Μελλοντικές επεκτάσεις

- ♦ Στόχος είναι να δημιουργηθεί μια καινοτόμα εφαρμογή που θα αντικαταστήσει το υπάρχων τετράδιο απουσιολόγιο.
- ♦ Θα δημιουργηθεί μια Web εφαρμογή διαχείρισης που θα παραδίδεται στο κάθε σχολείο μαζί με το APK της εφαρμογής.
Αυτή η εφαρμογή θα χρησιμοποιηθεί για την δημιουργήσει ο εκάστοτε διαχειριστής όλο το προφίλ του σχολείου που διαχειρίζεται και να περνάει τα δεδομένα του σχολείου στο σύστημα.
- ♦ Θα προσεγγιστούν σχολεία του εξωτερικού και θα ψηφιοποιήσουμε την καταγραφή των απουσιών τους.
- ♦ Θα δημιουργήσουμε ένα πρόγραμμα πληρωμών που θα προσφέρει την ηλεκτρονική μας πλατφόρμα ως υπηρεσία και θα στοχεύει στα σχολεία που δεν μπορούν να καλύψουν το ποσό της αγοράς πλήρους έκδοσης για την εγκατάσταση στις υποδομές τους.

Βιβλιογραφία

- ❖ React Native - Setup your environment
Σύνδεσμος: <https://reactnative.dev/docs/set-up-your-environment>
- ❖ React Native Documentation
Σύνδεσμος: <https://reactnative.dev/docs/getting-started>
- ❖ React Native - The Practical Guide [2024]
Σύνδεσμος: <https://www.udemy.com/course/react-native-the-practical-guide>
- ❖ Docker - Get started
Σύνδεσμος: <https://docs.docker.com/get-started/>
- ❖ Docker - PostgreSQL samples
Σύνδεσμος: <https://docs.docker.com/reference/samples/postgres/>
- ❖ Awesome Docker Compose samples Github repository
Σύνδεσμος: <https://github.com/docker/awesome-compose>
- ❖ Complete C# Masterclass
Σύνδεσμος: <https://www.udemy.com/course/complete-csharp-masterclass>
- ❖ Tutorial: Create a minimal API with ASP.NET Core
Σύνδεσμος: <https://learn.microsoft.com/en-us/aspnet/core/tutorials/min-web-api>

Λόγος δημιουργίας

Η απόφαση μου να δημιουργήσω το ψηφιακό απουσιολόγιο ως την πτυχιακή μου εργασία βασίστηκε στο πρόβλημα καταχώρισης των απουσιών από τα σχολεία στο σύστημα. Με αφορμή τις σχέσεις μου με τους καθηγητές του 5ου ΓΕΛ Βύρωνος, αποφάσισα ότι θα ήταν μια μοναδική ευκαιρία και για εμένα ως προγραμματιστή αλλά και του σχολείου να εξελιχθούμε παράλληλα.

1. **Γρηγορότερη καταγραφή απουσιών:** Η εφαρμογή επιτρέπει στους καθηγητές να καταγράφουν απουσίες με μεγαλύτερη ταχύτητα και ακρίβεια, μειώνοντας την ανάγκη για χειρόγραφες καταχωρήσεις. Με τη χρήση της εφαρμογής, η διαδικασία καταγραφής γίνεται άμεσα, ενώ μπορεί να γίνει ακόμα και κατά τη διάρκεια του μαθήματος. Αυτό εξοικονομεί πολύτιμο χρόνο για τους εκπαιδευτικούς και διασφαλίζει ότι οι απουσίες καταγράφονται χωρίς καθυστερήσεις ή λάθη.
2. **Ευκολότερη μεταφορά στο κεντρικό σύστημα MySchool:** Η εφαρμογή είναι σχεδιασμένη ώστε να διευκολύνει τη μεταφορά δεδομένων απουσιών στο κεντρικό σύστημα MySchool. Με ενσωματωμένες λειτουργίες εξαγωγής δεδομένων, οι καθηγητές και οι διοικητικοί υπάλληλοι μπορούν εύκολα να στείλουν τις καταγεγραμμένες απουσίες στο MySchool χωρίς επιπλέον βήματα. Αυτό απλοποιεί τη διαδικασία και μειώνει τον κίνδυνο λάθους

κατά τη μεταφορά των δεδομένων, καθιστώντας τη διαχείριση των απουσιών πιο αποτελεσματική και ασφαλή.