

ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΙΡΑΙΩΣ



ΣΧΟΛΗ ΤΕΧΝΟΛΟΓΙΩΝ ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ ΕΠΙΚΟΙΝΩΝΙΩΝ ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ

Πτυχιακή Εργασία

Τίτλος Διατριβής	«Verifiable Credentials: Εμβάθυνση στις νέες τεχνολογίες online αυθεντικοποίησης με Self Sovereign Identity. » «New techniques for SSI based remote presentation online. »
Ονοματεπώνυμο Φοιτητή	Κωνσταντίνος Σκλαβενίτης
Πατρώνυμο	Θωμάς
Αριθμός Μητρώου	Π21151
Επιβλέπων	Ευάγγελος Σακκόπουλος, Αναπληρωτής Καθηγητής

Πειραιάς, Ιούλιος 2025

Copyright ©

Απαγορεύεται η αντιγραφή, αποθήκευση και διανομή της παρούσας εργασίας, εξ ολοκλήρου ή τμήματος αυτής, για εμπορικό σκοπό. Επιτρέπεται η ανατύπωση, αποθήκευση και διανομή για σκοπό μη κερδοσκοπικό, εκπαιδευτικής ή ερευνητικής φύσης, υπό την προϋπόθεση να αναφέρεται η πηγή προέλευσης και να διατηρείται το παρόν μήνυμα.

Οι απόψεις και τα συμπεράσματα που περιέχονται σε αυτό το έγγραφο εκφράζουν αποκλειστικά τον συγγραφέα και δεν αντιπροσωπεύουν τις επίσημες θέσεις του Πανεπιστημίου Πειραιώς.

Ως συγγραφέας της παρούσας εργασίας δηλώνω πως η παρούσα εργασία δεν αποτελεί προϊόν λογοκλοπής και δεν περιέχει υλικό από μη αναφερόμενες πηγές.

Επιτελική Σύνοψη

Η παρούσα πτυχιακή εργασία εξετάζει τις σύγχρονες τεχνολογίες διαδικτυακής αυθεντικοποίησης χρηστών μέσω Verifiable Credentials (Επαληθεύσιμων Διαπιστευτηρίων) και του προτύπου Self-Sovereign Identity (SSI), με ιδιαίτερη έμφαση στο Ευρωπαϊκό οικοσύστημα ψηφιακής ταυτότητας - European Digital Identity (EUDI). Σκοπός της εργασίας είναι η θεωρητική και η τεχνική αναφορά στις αρχές, τα πρότυπα και τις υλοποιήσεις που διέπουν τις νέες μορφές ψηφιακής ταυτότητας και ταυτόχρονα να καταγράψει και να τεκμηριώσει την υλοποίηση ενός πλήρους σεναρίου αυθεντικοποίησης με βάση την έκδοση, αποθήκευση και επαλήθευση ενός διαπιστευτηρίου.

Κατά το θεωρητικό μέρος, η εργασία επικεντρώνεται στις βασικές έννοιες και τεχνολογίες που υποστηρίζουν τη νέα φιλοσοφία της αυτοκυριαρχούμενης ταυτότητας, όπως το SSI, τα Verifiable Presentations (VPs), τα Presentation Definitions (PD) και τα ερωτήματα DCQL, το πρότυπο OpenID for Verifiable Credentials (OpenID4VC), καθώς και στις λειτουργικές ροές και τις τεχνολογίες που συνθέτουν το EUDI Ecosystem.

Κατά το πρακτικό μέρος, εφαρμόστηκε ένας πλήρης κύκλος διαχείρισης ενός ψηφιακού διαπιστευτηρίου, με την ενσωμάτωση της Ελληνικής Ακαδημαϊκής Ταυτότητας στο οικοσύστημα του EUDI.

Πιο συγκεκριμένα:

- Επεκτάθηκε η λειτουργικότητα του Issuer (Εκδότη) από το επίσημο αποθετήριο κώδικα του EUDI, ώστε να εκδίδει ακαδημαϊκές ταυτότητες με βάση το αντίστοιχο Presentation Definition (ορισμός της δομής του διαπιστευτηρίου).
- Ο Issuer εγκαταστάθηκε τοπικά και έγινε χρήση reverse proxy για πρόσβαση και ενσωμάτωση στην Android εφαρμογή του ψηφιακού πορτοφολιού EUDI Wallet.
- Υλοποιήθηκε η διαδικασία παρουσίασης και επαλήθευσης του πιστοποιητικού μέσω μίας νέας διεπαφής χρήστη (frontend), η οποία χρησιμοποιεί το EUDI Backend Verifier Endpoint, με υποστήριξη για Presentation Definition.

Συνολικά, η εργασία αποσκοπεί στο να συνδέσει το θεωρητικό υπόβαθρο των αποκεντρωμένων ταυτοτήτων με μία υλοποίηση στα πλαίσια της ευρύτερης προσπάθειας για ψηφιακό μετασχηματισμό της ταυτότητας στην Ευρώπη.

Ευχαριστίες

Θα ήθελα να ευχαριστήσω θερμά τον επιβλέποντα καθηγητή μου, κύριο Ευάγγελο Σακκόπουλο, για την πολύτιμη καθοδήγησή του, την ευκαιρία που μου έδωσε να ασχοληθώ με το παρόν θέμα και την άψογη συνεργασία μας.

Ευχαριστώ επίσης όλους ανεξαιρέτως τους διδάσκοντες του Τμήματος Πληροφορικής για τη συμβολή τους στην ακαδημαϊκή και επιστημονική μου εξέλιξη.

Τέλος, εκφράζω την ευγνωμοσύνη μου στους κοντινούς μου ανθρώπους για τη στήριξη και την ενθάρρυνση που μου προσέφεραν καθ' όλη τη διάρκεια των σπουδών μου και της εκπόνησης της παρούσας εργασίας.

Ιούλιος 2025

Κωνσταντίνος Σκλαβενίτης

ΠΙΝΑΚΑΣ ΠΕΡΙΕΧΟΜΕΝΩΝ

1	Εισαγωγή.....	7
1.1	Περιγραφή του υπό μελέτη προβλήματος	7
1.2	Σκοπός και στόχοι της εργασίας.....	7
1.3	Παραδοτέα της εργασίας.....	8
1.4	Δομή της εργασίας	8
2	Θεωρητικό Υπόβαθρο & Τεχνολογίες.....	10
2.1	Self-Sovereign Identity (SSI)	10
2.2	Verifiable Credentials & Verifiable Presentations	11
2.3	OpenID for Verifiable Credentials (OpenID4VC)	13
2.4	Presentation Definition και DCQL – Τρόποι καθορισμού απαιτήσεων παρουσίασης	15
3	Το οικοσύστημα EUDI	17
3.1	Γενική Επισκόπηση.....	17
3.2	Ροή Διαπιστευτηρίων (Credentials Lifecycle).....	18
3.3	Επιμέρους Στοιχεία – Εφαρμογές.....	19
4	Υλοποίηση – Υποστήριξη Ελληνικής Ακαδημαϊκής Ταυτότητας στο Οικοσύστημα του EUDI... ..	22
4.1	Τροποποίηση και παραμετροποίηση του Εκδότη (Issuer) για έκδοση Ελληνικής Ακαδημαϊκής Ταυτότητας	23
4.2	Διαμόρφωση Reverse Proxy και προσαρμογή του Πορτοφολιού (Wallet) για έκδοση Ελληνικής Ακαδημαϊκής Ταυτότητας με χρήση του εκτεταμένου Εκδότη (Issuer)	28
4.3	Υλοποίηση Επαληθευτή (Verifier) Ελληνικών Ακαδημαϊκών Ταυτοτήτων	38
5	Συμπεράσματα.....	47
6	Βιβλιογραφικές Πηγές	48

Κεφάλαιο 1^ο

1 Εισαγωγή

1.1 Περιγραφή του υπό μελέτη προβλήματος

Η έννοια της ταυτότητας βρίσκεται στο επίκεντρο κάθε ψηφιακής συναλλαγής: από την πρόσβαση σε υπηρεσίες του Δημοσίου έως τη χρήση εφαρμογών στον ιδιωτικό τομέα. Η ανάγκη για αξιόπιστη, διαλειτουργική και ασφαλή ταυτοποίηση χρηστών στο διαδίκτυο είναι σήμερα πιο έντονη από ποτέ. Σε αυτό το πλαίσιο, η Ευρωπαϊκή Ένωση προχώρησε στον επανασχεδιασμό του θεσμικού πλαισίου ταυτοποίησης, εισάγοντας τον Κανονισμό eIDAS 2.0, που προβλέπει τη δημιουργία της Ευρωπαϊκής Ψηφιακής Ταυτότητας (European Digital Identity - EUDI) και τη διάθεση ψηφιακών πορτοφολιών σε όλους τους πολίτες της Ε.Ε.

Η μετάβαση από ομοσπονδιακά μοντέλα ταυτοποίησης σε αποκεντρωμένες μορφές όπου ο χρήστης έχει τον έλεγχο των προσωπικών του δεδομένων, αποτέλεσε το εφαλτήριο για την ανάπτυξη της αρχιτεκτονικής του Self-Sovereign Identity (SSI). Βασισμένο σε πρότυπα που καθορίζει το W3C, όπως τα Verifiable Credentials (VCs) και τα Verifiable Presentations (VPs), το μοντέλο αυτό επιδιώκει να μεταφέρει την «κυριαρχία» της ταυτότητας από τους παρόχους στον χρήστη.

Ωστόσο, η ενσωμάτωση αυτής της προσέγγισης σε ένα λειτουργικό οικοσύστημα απαιτεί τη συνύπαρξη πολλών τεχνολογιών. Το EUDI Wallet, ως μηχανισμός υλοποίησης αυτής της ευρωπαϊκής προσπάθειας, ενώνει όλες αυτές τις τεχνολογίες και πρότυπα σε ένα ενιαίο σύστημα.

Παρά τα θετικά βήματα, τα εργαλεία και τα πρότυπα αυτά είναι ακόμα σε εξέλιξη και απαιτούνται αρκετές προσαρμογές τόσο σε δημόσια, όσο και ιδιωτικά συστήματα, κάτι που επιχειρεί να προσεγγίσει η παρούσα εργασία, με έμφαση στην ένταξη της Ελληνικής Ακαδημαϊκής Ταυτότητας στο οικοσύστημα του EUDI.

1.2 Σκοπός και στόχοι της εργασίας

Ο σκοπός της παρούσας πτυχιακής εργασίας είναι διπλός. Αρχικά να αναλύσει σε θεωρητικό επίπεδο τις τεχνολογίες, τα πρότυπα και τα θεσμικά πλαίσια που σχετίζονται με την ψηφιακή ταυτοποίηση μέσω Self-Sovereign Identity και Verifiable Credentials, και δεύτερον να υλοποιήσει στην πράξη ένα σενάριο έκδοσης, αποθήκευσης και επαλήθευσης ενός διαπιστευτηρίου.

Η εργασία επικεντρώνεται στη σύνδεση μεταξύ θεωρίας και πράξης. Μέσα από την εμβάθυνση σε πρότυπα όπως τα VC, VP, OpenID4VP, Presentation Definitions και DCQL, καθώς και την ανάλυση των λειτουργικών ροών του EUDI (Issuer, Wallet, Verifier),

επιδιώκεται η κατανόηση του τρόπου με τον οποίο συγκροτείται ένα σύγχρονο σύστημα ψηφιακής ταυτότητας.

Οι στόχοι της εργασίας συνοψίζονται ως εξής:

1. Η μελέτη των βασικών εννοιών της αποκεντρωμένης ψηφιακής ταυτότητας, με επίκεντρο το μοντέλο του SSI και τα πρότυπα Verifiable Credentials και Verifiable Presentations.
2. Η διερεύνηση και κατανόηση τεχνολογιών και πρωτοκόλλων που εφαρμόζονται στο EUDI, όπως το OpenID4VC.
3. Η παρουσίαση του οικοσυστήματος EUDI και των βασικών του συστατικών (Issuer, Wallet, Verifier).
4. Η υλοποίηση ενός σεναρίου χρήσης για την Ελληνική Ακαδημαϊκή Ταυτότητα, με έκδοση, παρουσίαση και επαλήθευση του σχετικού διαπιστευτηρίου.

1.3 Παραδοτέα της εργασίας

Στο πλαίσιο προσέγγισης του υπό μελέτης προβλήματος, τα παραδοτέα της εργασίας περιλαμβάνουν τα εξής:

1. Έντυπο κείμενο τεκμηρίωσης της πτυχιακής εργασίας:

Αποτελεί το παρόν έγγραφο και περιέχει την περιγραφή τόσο του θεωρητικού όσο και του πρακτικού σκέλους της εργασίας, καθώς και προσωπικές σκέψεις και αναλύσεις σχετικά με το αντικείμενο θέμα της.

2. Το λογισμικό που αναπτύχθηκε ή τροποποιήθηκε κατά το πρακτικό σκέλος της εργασίας:

Βρίσκεται σε αποθετήρια κώδικα στην πλατφόρμα του GitHub, τα οποία είναι χωρισμένα σε τμήματα ανάλογα με τη λειτουργία και την εφαρμογή τους. Είναι συγκεντρωμένα στο κεντρικό αποθετήριο: <https://github.com/konsklav/gracid-eudiw-thesis>

3. Συλλογή πηγών και βιβλιογραφία:

Η εργασία συνοδεύεται από την βιβλιογραφία από την οποία αντλήθηκαν οι ερμηνείες και οι ορισμοί των εννοιών, οι πληροφορίες των υπό μελέτη προβλημάτων καθώς και πηγές που συνετέλεσαν στην πρακτική υλοποίηση του θέματος τόσο σε επίπεδο εκπαιδευτικό όσο και ανάπτυξης και επίλυσης τεχνικών ζητημάτων.

1.4 Δομή της εργασίας

Η παρούσα εργασία είναι οργανωμένη σε έξι (6) κεφάλαια, με στόχο τη σταδιακή παρουσίαση του θεωρητικού υποβάθρου, την τεκμηρίωση της πρακτικής υλοποίησης και την αξιολόγηση των αποτελεσμάτων. Συγκεκριμένα:

- Στο **Κεφάλαιο 1** παρουσιάζεται η εισαγωγή στο θέμα, ορίζεται το υπό μελέτη πρόβλημα, αναλύονται ο σκοπός και οι στόχοι της εργασίας, προσδιορίζονται τα παραδοτέα και περιγράφεται η συνολική δομή του εγγράφου.

- Στο **Κεφάλαιο 2** αναλύεται το θεωρητικό τεχνικό υπόβαθρο που στηρίζει την υλοποίηση. Παρουσιάζονται οι βασικές έννοιες του προτύπου Self-Sovereign Identity (SSI), τα πρότυπα Verifiable Credentials (VCs) και Verifiable Presentations (VPs), το πρωτόκολλο OpenID4VC, καθώς και οι δομές PD και DCQL.
- Στο **Κεφάλαιο 3** περιγράφεται το οικοσύστημα της Ευρωπαϊκής Ψηφιακής Ταυτότητας (European Digital Identity - EUDI), τα κύρια συστατικά του (Issuer, Wallet, Verifier) και η λειτουργική ροή των διαπιστευτηρίων εντός αυτού.
- Στο **Κεφάλαιο 4** παρουσιάζεται και τεκμηριώνεται η πρακτική υλοποίηση της εργασίας. Περιγράφεται η διαδικασία τροποποίησης του Issuer, η ενσωμάτωσή του στο EUDI Wallet και η δημιουργία custom Verifier UI για χρήση σε συνδυασμό με τον υπάρχοντα online επαληθευτή του EUDI οικοσυστήματος. Παρουσιάζεται επίσης αναλυτικά η λειτουργική ροή των συστημάτων αυτών με παραδείγματα χρήσης και ενδεικτικά στιγμιότυπα της οθόνης.
- Στο **Κεφάλαιο 5** καταγράφονται τα συμπεράσματα και η αξιολόγηση της εργασίας. Παρουσιάζεται η κριτική αποτίμηση τόσο της θεωρητικής ανάλυσης όσο και της πρακτικής υλοποίησης.
- Στο **Κεφάλαιο 6** παρατίθενται οι βιβλιογραφικές πηγές που αξιοποιήθηκαν για την ανάπτυξη της παρούσας εργασίας.

Η δομή αυτή επιτρέπει στον αναγνώστη να αποκτήσει μία σφαιρική εικόνα του θέματος, από την αρχική θεωρητική θεμελίωση μέχρι και την εφαρμογή σε ένα πραγματικό σενάριο χρήσης.

Κεφάλαιο 2^ο

2 Θεωρητικό Υπόβαθρο & Τεχνολογίες

2.1 Self-Sovereign Identity (SSI)

Το Self-Sovereign Identity αποτελεί μια προσέγγιση διαχείρισης ψηφιακής ταυτότητας, κατά την οποία το άτομο έχει τον πλήρη έλεγχο των προσωπικών του πληροφοριών και της διαδικασίας αυθεντικοποίησης. Αντί να βασίζεται σε κεντρικούς ή ομοσπονδιακούς παρόχους ταυτότητας, το μοντέλο του SSI επιτρέπει στους χρήστες να αποθηκεύουν, να διαχειρίζονται και να παρουσιάζουν οι ίδιοι τα αναγνωριστικά τους στοιχεία (credentials) απευθείας σε εφαρμογές, οργανισμούς ή υπηρεσίες.

Σε αντίθεση με τα κεντρικά συστήματα ταυτότητας όπου ο πάροχος (π.χ. Google, Facebook) ελέγχει τα δεδομένα του χρήστη, το SSI επιτρέπει στον ίδιο τον χρήστη να έχει τον πλήρη έλεγχο των προσωπικών του πληροφοριών και να αποφασίζει πότε, σε ποιόν και πόσα από αυτά θα κοινοποιήσει, χωρίς να απαιτείται η μεσολάβηση τρίτων.

(Wikipedia – SSI, 2025)

Τα συστήματα αυτοκυριαρχούμενης ταυτότητας οφείλουν να ακολουθούν ένα σύνολο αρχών που διασφαλίζουν την ιδιωτικότητα, τη διαλειτουργικότητα, την προσβασιμότητα, καθώς και την ακεραιότητα και αυθεντικότητα της ταυτότητας. Οι αρχές αυτές παρουσιάζονται συγκεντρωτικά από το Sovrin Foundation στο πλαίσιο του εγγράφου Principles of SSI v3 (Sovrin, 2022):

- Υποκειμενικότητα και Ύπαρξη: Κάθε άτομο ή οντότητα έχει το δικαίωμα να διαθέτει ταυτότητα που δεν εξαρτάται από κάποιον εξωτερικό πάροχο ή εξουσιοδότηση τρίτων.
- Αυτονομία και Έλεγχος: Οι χρήστες έχουν τον πλήρη έλεγχο των ταυτοτήτων και των διαπιστευτηρίων τους, καθώς και της χρήσης τους.
- Πρόσβαση και Διαφάνεια: Οι χρήστες έχουν άμεση, πλήρη και κατανοητή πρόσβαση στα δεδομένα τους και στους κανόνες που τα διέπουν.
- Ασφάλεια και Αυθεντικότητα: Τα τεχνικά μέσα που χρησιμοποιούνται πρέπει να διασφαλίζουν την ακρίβεια, την ακεραιότητα και την επαληθευσσιμότητα της ταυτότητας.
- Ελαχιστοποίηση και Ιδιωτικότητα: Τα συστήματα πρέπει να επιτρέπουν την κοινοποίηση μόνο των απαραίτητων πληροφοριών για κάθε χρήση, με δυνατότητα επιλεκτικής αποκάλυψης.
- Φορητότητα και Διαλειτουργικότητα: Οι ταυτότητες πρέπει να είναι επαναχρησιμοποιήσιμες σε διαφορετικά πλαίσια και να υποστηρίζονται από ανοιχτά, καθολικά αποδεκτά πρότυπα.

- **Αδιάλειπτη Χρήση και Διατήρηση:** Οι ταυτότητες πρέπει να διατηρούνται ανεξάρτητα από μεμονωμένες εφαρμογές ή παρόχους, και να παραμένουν λειτουργικές με την πάροδο του χρόνου.
- **Συγκατάθεση:** Ο κάτοχος της ταυτότητας πρέπει να έχει την πρωτοβουλία και επιλογή για κάθε χρήση, παρουσίαση ή κοινοποίηση των προσωπικών του δεδομένων.
- **Καθολική Συμμετοχή και Δίκαιη Πρόσβαση:** Το σύστημα πρέπει να είναι προσβάσιμο τεχνικά και κοινωνικά, χωρίς διακρίσεις, αποκλεισμούς ή τεχνολογικά εμπόδια.
- **Ανοιχτότητα και Λογοδοσία:** Η ανάπτυξη και η χρήση τεχνολογιών SSI πρέπει να γίνεται σε πλαίσιο διαφάνειας και με δυνατότητα λογοδοσίας για κάθε εμπλεκόμενο.

Η φιλοσοφία του SSI αντανακλά μια νέα ηθική στο πεδίο της ψηφιακής ταυτότητας, όπου ο χρήστης δεν είναι αποδέκτης, αλλά διαχειριστής των προσωπικών του δεδομένων. Αυτή η αποκεντρωμένη προσέγγιση επιχειρεί να επαναφέρει την εμπιστοσύνη στο ψηφιακό περιβάλλον, ελαχιστοποιώντας τον κίνδυνο κεντρικών παραβιάσεων δεδομένων και ενισχύοντας τη λογοδοσία μέσω τεχνολογίας.

2.1.1 Ζητήματα Ασφάλειας και Κανονιστικής Συμμόρφωσης

Το μοντέλο SSI ενσωματώνει βασικές αρχές προστασίας προσωπικών δεδομένων, όπως η ελαχιστοποίηση πληροφορίας και η κυριότητα των δεδομένων από τον ίδιο τον χρήστη, ευθυγραμμισμένο με τον Γενικό Κανονισμό για την Προστασία Δεδομένων (GDPR) της Ε.Ε. (European Union, 2016).

Παράλληλα, ο αναθεωρημένος κανονισμός eIDAS 2.0 αναγνωρίζει την έννοια του Ευρωπαϊκού Ψηφιακού Πορτοφολιού (EUDIW), απαιτώντας την υιοθέτηση τεχνικών μέτρων όπως ισχυρή κρυπτογράφηση, αποθήκευση των κλειδιών σε τοπικές συσκευές και μηχανισμούς πολυπαραγοντικής ταυτοποίησης (European Commission, 2024).

Ήδη από το 2019, η Ευρωπαϊκή Επιτροπή είχε υπογραμμίσει ότι η φιλοσοφία του SSI μπορεί να υποστηριχθεί θεσμικά, ανοίγοντας τον δρόμο για την ενσωμάτωσή του στο Ευρωπαϊκό νομικό και τεχνικό οικοσύστημα μέσω του eIDAS (European Commission, 2019).

2.2 Verifiable Credentials & Verifiable Presentations

Τα Verifiable Credentials και Verifiable Presentations αποτελούν βασικά δομικά στοιχεία του μοντέλου ψηφιακής ταυτότητας SSI, όπως ορίζονται από το W3C.

2.2.1 Verifiable Credentials (VCs)

Τα VCs είναι ψηφιακά διαπιστευτήρια που περιέχουν ένα σύνολο δεδομένων ισχυρισμών (*claims*) για ένα υποκείμενο. Συνοδεύονται από μεταδεδομένα όπως ο εκδότης (*issuer*), η ημερομηνία έκδοσης (*issuanceDate*), η ημερομηνία λήξης

(*expirationDate*) και μία κρυπτογραφική απόδειξη αυθεντικότητας, συνήθως με τη μορφή ψηφιακής υπογραφής (W3C, 2023).

Η δομή τους έχει σχεδιαστεί έτσι, ώστε να είναι tamper-evident, δηλαδή οποιαδήποτε τροποποίηση του περιεχομένου τους να είναι ανιχνεύσιμη, διατηρώντας έτσι την ακεραιότητα των δεδομένων. Επιπλέον, είναι πλήρως επεκτάσιμα για προσαρμογή σε συγκεκριμένες ανάγκες εφαρμογών (W3C, 2023).

Τα VCs υποστηρίζουν επιλεκτική αποκάλυψη (*selective disclosure*), επιτρέποντας στον κάτοχο να κοινοποιήσει μόνο συγκεκριμένες πληροφορίες και όχι το πλήρες σύνολο του διαπιστευτηρίου, ακολουθώντας την αρχή της ελαχιστοποίησης των δεδομένων.

Το μοντέλο εμπιστοσύνης που υιοθετείται βασίζεται στην τριάδα:

- Issuer: Εκδίδει το VC.
- Holder: Κατέχει και διαχειρίζεται το VC.
- Verifier: Ελέγχει την εγκυρότητα του VC που του παρουσιάζεται.

Κάθε μέρος εμπιστεύεται τον Issuer με βάση μηχανισμούς όπως trust frameworks ή λίστες έγκυρων εκδοτών (W3C, 2023; Wikipedia – Verifiable Credentials, 2025).

Τα VCs βασίζονται σε μορφή **JSON-LD** (linked data) και μπορούν να υποστηρίζονται από διάφορα πρότυπα κρυπτογράφησης και υπογραφής, όπως **JWS**, **SD-JWT**, και **COSE** (W3C, 2023).

2.2.2 Verifiable Presentations (VPs)

Τα VPs είναι δομές που περιλαμβάνουν ένα ή περισσότερα VCs, τα οποία ο κάτοχος παρουσιάζει σε έναν επαληθευτή (Verifier) κατά τη διαδικασία αυθεντικοποίησης ή πρόσβασης σε υπηρεσία.

Ένα VP περιέχει:

- Μεταδεδομένα
- Ένα ή περισσότερα VCs
- Μία ή περισσότερες αποδείξεις (proofs), π.χ. ψηφιακές υπογραφές

Η χρήση VPs επιτρέπει στον κάτοχο να εφαρμόσει επιλεκτική αποκάλυψη κοινοποιώντας μόνο τα απολύτως απαραίτητα στοιχεία για τη συγκεκριμένη συναλλαγή. Επιπλέον, δίνεται η δυνατότητα χρήσης παραστάσεων τύπου predicate, όπως π.χ. “ηλικία > 18”, χωρίς αποκάλυψη του ακριβούς αριθμού (W3C, 2023).

Τα VPs συνήθως είναι προσωρινές δομές, μίας χρήσης, κατάλληλες για επαλήθευση σε συγκεκριμένες συναλλαγές, ενισχύοντας την προστασία της ιδιωτικότητας (W3C, 2023; Wikipedia – Verifiable Credentials, 2025).

2.2.3 Ζητήματα Ασφαλείας

Η ασφάλεια των VCs και VPs διασφαλίζεται μέσω της χρήσης:

- Ψηφιακών υπογραφών όπως **JWS (JSON Web Signature)**,
- Κρυπτογραφικών τεχνικών όπως **SD-JWT (Selective Disclosure JWT)**,

- Προτύπων lightweight κρυπτογράφησης όπως **COSE (CBOR Object Signing and Encryption)**

Αυτά τα πρότυπα εξασφαλίζουν:

- Την ακεραιότητα του περιεχομένου του διαπιστευτηρίου,
- Την αυθεντικότητα του εκδότη,
- Την προστασία της ιδιωτικότητας.

Σύμφωνα με το Verifiable Credentials Data Model 2.0, η επαλήθευση ενός VC ή VP δεν απαιτεί άμεση επικοινωνία μεταξύ του Issuer και του Verifier, γεγονός που ενισχύει την αυτονομία του χρήστη και μειώνει τον κίνδυνο παρακολούθησης ή αποκάλυψης περιττών πληροφοριών (W3C, 2023).

2.3 OpenID for Verifiable Credentials (OpenID4VC)

Το OpenID for Verifiable Credentials (OpenID4VC) είναι ένας γενικός όρος που αναφέρεται στο σύνολο των προδιαγραφών της OpenID Foundation για την υποστήριξη της έκδοσης και παρουσίασης Verifiable Credentials με χρήση του πρωτοκόλλου OpenID Connect (OIDC) (OpenID Foundation, 2025).

Αποτελείται κυρίως από δύο επιμέρους προδιαγραφές:

- OpenID4VCI – για την έκδοση διαπιστευτηρίων από έναν εκδότη (issuer) προς ένα ψηφιακό πορτοφόλι (wallet).
- OpenID4VP – για την παρουσίαση αυτών των διαπιστευτηρίων από το wallet σε έναν επαληθευτή (verifier).

Η οικογένεια OpenID4VC έχει σχεδιαστεί ώστε να προσφέρει συμβατότητα με υπάρχουσες υποδομές OIDC/OAuth 2.0, αξιοποιώντας μηχανισμούς και flows που είναι ήδη γνωστά στον χώρο της ταυτοποίησης. Με αυτόν τον τρόπο, οργανισμοί μπορούν να υιοθετήσουν τεχνολογίες αυτοκυριαρχούμενης ταυτότητας χωρίς να εγκαταλείψουν τις υπάρχουσες πλατφόρμες τους (OpenID Foundation, 2025).

Επιπλέον, το OpenID4VC λειτουργεί ως γέφυρα μεταξύ της παραδοσιακής αυθεντικοποίησης (π.χ. login με Google) και των αποκεντρωμένων μοντέλων ταυτοτήτων, προσφέροντας ευελιξία, ασφάλεια και επεκτασιμότητα. Είναι format-agnostic, καθώς υποστηρίζει πολλαπλά είδη διαπιστευτηρίων όπως W3C Verifiable Credentials, ISO mDL, SD-JWT VCs, και AnonCreds, ενώ διασφαλίζει κρυπτογραφικό δέσιμο στον κάτοχο και προστασία της ιδιωτικότητας.

Συνολικά, το OpenID4VC αποτελεί τη θεμελιώδη βάση για την υλοποίηση διαλειτουργικών και ασφαλών λύσεων ψηφιακής ταυτοποίησης, σύμφωνα με τις αρχές της αυτοκυριαρχούμενης ταυτότητας (SSI).

2.3.1 OpenID for Verifiable Credentials Issuance (OpenID4VCI)

Το OpenID for Verifiable Credential Issuance (OpenID4VCI) είναι μια προδιαγραφή που επεκτείνει το OpenID Connect (OIDC) για να υποστηρίξει την ασφαλή και διαλειτουργική έκδοση Verifiable Credentials από έναν πάροχο διαπιστευτηρίων

(issuer) προς ένα ψηφιακό πορτοφόλι (wallet), στο πλαίσιο του οικοσυστήματος των αποκεντρωμένων ταυτοτήτων (SSI) (OpenID Foundation, 2025).

2.3.1.1 Λειτουργία

- Ο χρήστης (μέσω της εφαρμογής του ψηφιακού πορτοφολιού - wallet) ξεκινά τη διαδικασία έκδοσης είτε μέσω pre-authorized flow είτε μέσω authorization code flow, ζητώντας ένα ή περισσότερα credentials από τον issuer (OpenID Foundation, 2025).
- Ο issuer παρέχει ένα credential offer URI, μέσω του οποίου το wallet λαμβάνει πληροφορίες σχετικά με τα διαθέσιμα credentials, τον τύπο τους και τις προϋποθέσεις (W3C CCG, 2023).
- Το wallet ενεργεί ως OIDC client και διαπραγματεύεται την έκδοση με τον issuer, στέλνοντας τις απαραίτητες αποδείξεις ταυτοποίησης ή attributes (π.χ. DID, proof of possession).
- Ο issuer ελέγχει τα δεδομένα, εκδίδει το κατάλληλο Verifiable Credential και το παραδίδει στο wallet με ασφάλεια (OpenID Foundation, 2025).

Συνοψίζοντας, το OpenID4VCI προσφέρει έναν ασφαλή, τυποποιημένο και ευέλικτο τρόπο για την έκδοση Verifiable Credentials, αποτελώντας βασικό στοιχείο για την υλοποίηση της αποκεντρωμένης ταυτοποίησης σε συμβατές και παραμετροποιήσιμες υποδομές.

2.3.2 OpenID for Verifiable Presentations (OpenID4VP)

Το OpenID for Verifiable Presentations (OpenID4VP) είναι μια προδιαγραφή που επεκτείνει το πρωτόκολλο OpenID Connect (OIDC), με στόχο την ασφαλή και δομημένη παρουσίαση Verifiable Presentations από ένα ψηφιακό πορτοφόλι (wallet) σε έναν επαληθευτή (verifier), στο πλαίσιο αποκεντρωμένων ψηφιακών ταυτοτήτων (SSI).

(Wikipedia)

2.3.2.1 Λειτουργία

- Ο Verifier (επαληθευτής) ξεκινά τη διαδικασία, δημιουργώντας ένα αίτημα (authorization request) όπου ζητά συγκεκριμένα διαπιστευτήρια από τον χρήστη. Το αίτημα αυτό περιέχει έναν τύπο απάντησης vp_token, αντί του κλασικού id_token.
- Ο χρήστης χρησιμοποιεί το wallet του για να διαχειριστεί το αίτημα. Το wallet λειτουργεί σαν OIDC Authorization Server και είναι υπεύθυνο να δημιουργήσει ένα κατάλληλο vp_token που περιέχει τα Verifiable Presentations, μαζί με αποδείξεις (proofs) για την εγκυρότητα και την κυριότητα των διαπιστευτηρίων.
- Το wallet στέλνει την απάντηση είτε μέσω redirect αν βρίσκεται στην ίδια συσκευή με τον Verifier (same-device flow), είτε μέσω HTTP POST αν βρίσκεται σε διαφορετική συσκευή (cross-device flow).
- Ο Verifier λαμβάνει το vp_token, το επεξεργάζεται, επαληθεύει τα credentials, και ολοκληρώνει τη συναλλαγή ή την ταυτοποίηση.

(OpenID Foundation, 2025)

Συνοπτικά, το OpenID4VP προσφέρει μια συνεκτική, ασφαλή και ευπροσάρμοστη μέθοδο για την παρουσίαση Verifiable Presentations, αποτελώντας κρίσιμο συστατικό για την Self-Sovereign Identity υλοποίηση.

2.4 Presentation Definition και DCQL – Τρόποι καθορισμού απαιτήσεων παρουσίασης

Κατά την παρουσίαση Verifiable Credentials στο πλαίσιο πρωτοκόλλων όπως το OID4VP, ο επαληθευτής (verifier) χρειάζεται έναν τυποποιημένο τρόπο για να δηλώσει ποια διαπιστευτήρια ζητά από τον χρήστη.

Σήμερα, υπάρχουν δύο βασικές μέθοδοι για αυτή τη λειτουργία:

- i. Το Presentation Definition (PD), που βασίζεται σε προκαθορισμένες δομές JSON schema, και
- ii. Η Digital Credential Query Language (DCQL), που επιτρέπει πιο δυναμικά και εκφραστικά ερωτήματα, βασισμένα σε λογικές συνθήκες επί των δεδομένων, δίνοντας μεγαλύτερη ευελιξία στην πλευρά του verifier.

2.4.1 Presentation Definition (PD)

Το Presentation Definition είναι μια από τις πιο διαδεδομένες μεθόδους περιγραφής απαιτήσεων παρουσίασης. Αναπτύχθηκε από την Decentralized Identity Foundation (DIF) και αποτελεί το κύριο μέσο δόμησης αιτημάτων στο πλαίσιο του Presentation Exchange v2.0 (DIF, 2023).

Χαρακτηριστικά:

- Είναι ένα JSON schema που περιγράφει με ακρίβεια τι είδους Verifiable Credentials απαιτείται να υποβληθούν.
- Περιλαμβάνει input_descriptors, constraints, fields, καθώς και φίλτρα σε attributes, issuers, formats (Walt.id, 2025).
- Χρησιμοποιείται ευρέως σε OID4VP ροές, με το πεδίο presentation_definition ή presentation_definition_uri.

Παρέχει καλό έλεγχο για τα credentials, αλλά συχνά χαρακτηρίζεται από πολυπλοκότητα και περιορισμένη εκφραστικότητα.

2.4.2 DCQL - Digital Credential Query Language

Η DCQL εισήχθη στο OpenID4VP Draft 22 (Οκτώβριος 2024) και συνεχίζει να εξελίσσεται. Πρόκειται για έναν εναλλακτικό, πιο απλό τρόπο με τον οποίο ο verifier μπορεί να καθορίσει τις απαιτήσεις του (OpenID Foundation, 2025).

Αντί για schema, η DCQL βασίζεται σε JSON δομές με φίλτρα και περιορισμούς (constraints) σε επίπεδο claims (ισχυρισμών). Η κύρια δομή είναι το credential_query, όπου περιγράφονται:

- Ο επιθυμητός τύπος διαπιστευτηρίου
- Οι έγκυροι εκδότες (trusted_issuers)
- Λίστα από claims και claims-level constraints.

Η DCQL είναι εξαιρετικά εκφραστική, πιο lightweight από το PD και πιο κατάλληλη για δυναμικές και αυτοματοποιημένες πλατφόρμες παρουσίασης (OpenID Foundation, 2025).

Λόγω της πρόσφατης δημιουργίας της, η ενσωμάτωσή της σε συστήματα επαλήθευσης βρίσκεται ακόμα υπό ανάπτυξη.

Κεφάλαιο 3^ο

3 Το οικοσύστημα EUDI

3.1 Γενική Επισκόπηση

Το European Digital Identity (EUDI) αποτελεί την επίσημη τεχνική πρόταση της Ευρωπαϊκής Ένωσης για τη δημιουργία ενός ενιαίου, ασφαλούς και διαλειτουργικού συστήματος ψηφιακής ταυτότητας, με νομική ισχύ σε όλα τα κράτη-μέλη της Ε.Ε. Βασίζεται στον αναθεωρημένο κανονισμό eIDAS 2.0 (Κανονισμός ΕΕ 2024/1183), ο οποίος εισάγει το δικαίωμα κάθε πολίτη να αποκτήσει ένα ψηφιακό πορτοφόλι (wallet) για την αποθήκευση, παρουσίαση και διαχείριση προσωπικών διαπιστευτηρίων (European Commission, 2024).

Το EUDI στηρίζεται στις αρχές της αυτοκυριαρχούμενης ταυτότητας και αξιοποιεί πρότυπα όπως τα Verifiable Credentials, τα Verifiable Presentations και το πρωτόκολλο OpenID4VP.

Η βασική του λειτουργία βασίζεται σε μια τριμερή σχέση εμπιστοσύνης μεταξύ:

- Issuer (Εκδότη): Φορέας που εκδίδει ψηφιακά διαπιστευτήρια με κρυπτογραφική υπογραφή (π.χ. πανεπιστήμια, δημόσιες αρχές).
- Holder (Κάτοχο): Το φυσικό πρόσωπο που λαμβάνει και διαχειρίζεται τα διαπιστευτήριά του μέσω του Wallet.
- Verifier (Επαληθευτή): Οργανισμός ή υπηρεσία που ζητά την παρουσίαση ενός ή περισσότερων διαπιστευτηρίων και επαληθεύει την εγκυρότητά τους.

Η διακίνηση των διαπιστευτηρίων πραγματοποιείται μέσω ροών παρουσίασης (presentation flows), κατά τις οποίες ο κάτοχος επιλέγει τα στοιχεία που θα αποκαλύψει, με στόχο την ελαχιστοποίηση των δεδομένων που ανταλλάσσονται και τη συμμόρφωση με το GDPR.

Η τεχνική υλοποίηση του οικοσυστήματος βασίζεται σε ένα σύνολο λογισμικών συστατικών ανοιχτού κώδικα, τα οποία φιλοξενούνται και συντηρούνται στο επίσημο αποθετήριο κώδικα της Ευρωπαϊκής Επιτροπής στο GitHub.

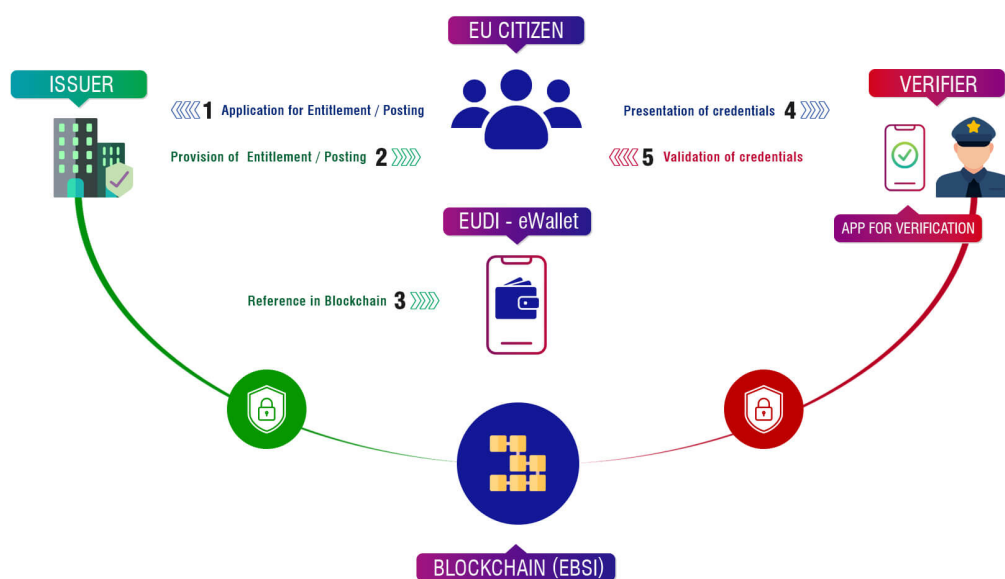
Οι βασικές συνιστώσες περιλαμβάνουν:

- Υπηρεσία έκδοσης διαπιστευτηρίων (Issuer Backend) - Εκδίδει Verifiable Credentials σε πολίτες με ψηφιακή υπογραφή.
- Εφαρμογή Wallet (Android / iOS) - Λειτουργεί ως φορητός αποθηκευτικός χώρος και interface για την παρουσίαση των credentials.
- Endpoint επαλήθευσης (Verifier Backend service) - Ζητά και επαληθεύει τα credentials που παρέχει ο χρήστης.

- Trust framework - Περιλαμβάνει κανόνες, μεταδεδομένα και απαιτήσεις πιστοποίησης για καθοδήγηση (συνεργασία) των επιμέρους μερών μέσω ασφαλών APIs και καθορισμένων flows.

Η ενιαία αρχιτεκτονική επιτρέπει την υλοποίηση αποκεντρωμένων σεναρίων αυθεντικοποίησης, χωρίς την ανάγκη για συνεχή εξάρτηση από κεντρικούς παρόχους ταυτότητας.

(EUDI – GitHub, 2025)



Εικόνα 3.1.1 – Ενδεικτική ροή ψηφιακού διαπιστευτηρίου στο οικοσύστημα EUDI, dc4eu.eu

3.2 Ροή Διαπιστευτηρίων (Credentials Lifecycle)

Η λειτουργία του EUDIW μπορεί να περιγραφεί ως μια συνδεδεμένη ροή τεσσάρων σταδίων, γνωστή και ως Credential Lifecycle:

1. Έκδοση (Issuance): Ο Issuer δημιουργεί και εκδίδει ένα VC στο wallet του χρήστη, μέσω OpenID Connect-based issuance flow.
2. Αποθήκευση (Storage): Το VC αποθηκεύεται τοπικά στο πορτοφόλι, με χρήση κρυπτογραφίας και on-device key protection.
3. Παρουσίαση (Presentation): Το πορτοφόλι δημιουργεί ένα VP με βάση τα αιτήματα του Verifier και τις πολιτικές παρουσίασης (presentation policies) (π.χ. PD ή DCQL).
4. Επαλήθευση (Verification): Ο Verifier επαληθεύει το VP, ελέγχει την υπογραφή, την εγκυρότητα του issuer και την αντιστοίχιση με το αίτημα.

(EUDI – GitHub, 2025)

3.3 Επιμέρους Στοιχεία – Εφαρμογές

3.3.1 Εκδότης (Issuer)

Ο Issuer αποτελεί ένα παραμετροποιήσιμο backend σύστημα για την έκδοση Verifiable Credentials στο πλαίσιο του EUDIW. Αναπτύσσεται σε Python, βασισμένο στο Flask με υποστήριξη για RESTful endpoints και υλοποιεί πλήρως την ασφαλή έκδοση διαπιστευτηρίων σε μορφή JWT, SD-JWT, W3C JSON-LD και COSE, μέσω τυποποιημένων ροών με χρήση authorization tokens ή pre-authorized codes. Η έκδοση βασίζεται στο πρότυπο OpenID for Verifiable Credential Issuance (OpenID4VCI), το οποίο επιτρέπει στον issuer να επικυρώνει την ταυτότητα του χρήστη και να του μεταδίδει το διαπιστευτήριο με ασφαλή και τυποποιημένο τρόπο.

Τεχνικά Χαρακτηριστικά:

- Παρέχει metadata endpoint (/well-known/openid-credential-issuer), το οποίο δημοσιεύει τις ικανότητες του issuer (supported formats, credential types, encryption keys, κ.ά.).
- Χρησιμοποιεί JWK (JSON Web Key) Sets για την έκθεση των δημόσιων κλειδιών του issuer, μέσω endpoint (/credential-issuer/well-known/jwks.json), ώστε να μπορούν τα wallets να επαληθεύσουν τις υπογραφές.
- Υποστηρίζει ενσωμάτωση με trust registries, metadata services και federation policies, για να διασφαλίζει ότι ο issuer είναι αναγνωρίσιμος και έγκυρος στο οικοσύστημα του EUDI.
- Περιλαμβάνει σύστημα credential_offer URIs, ώστε τα wallets να μπορούν να ξεκινούν το flow βάσει μίας «προσφοράς» (offer).
- Δομή credentials: Περιλαμβάνει δυναμική υποστήριξη για πολλαπλούς τύπους credentials, με έλεγχο schema, issuance policies και presentation definitions. Η συγκεκριμένη υλοποίηση επιτρέπει την παραμετροποίηση για custom credential types (π.χ. Ελληνική Ακαδημαϊκή Ταυτότητα).

(EUDI – GitHub, 2025; European Commission, 2025)

3.3.2 Ψηφιακό Πορτοφόλι (Digital Wallet)

Το EUDI Wallet είναι μια mobile εφαρμογή (Android/iOS) που επιτρέπει στον πολίτη να λαμβάνει (μέσω του Issuer), αποθηκεύει και παρουσιάζει (στον Verifier) με ασφάλεια Verifiable Credentials (π.χ. ψηφιακά διαβατήρια, ταυτότητες, άδειες οδήγησης, ψηφιακά έγγραφα) χρησιμοποιώντας το πρότυπο OpenID4VC. (EUDI - ARF, 2025)

Είναι σχεδιασμένο ως ένα σύγχρονο mobile σύστημα, του οποίου η ανάπτυξη είναι καταναμεμημένη σε διακριτές ενότητες, καθεμία υπεύθυνη για διαφορετική λειτουργία. Το frontend έχει αναπτυχθεί σε Kotlin για την Android εφαρμογή και Swift για την αντίστοιχη iOS. Ο «πυρήνας» (backend) του πορτοφολιού (όπως η αποθήκευση διαπιστευτηρίων, η διαχείριση συνεδρίας και οι ροές επικοινωνίας με

εκδότες και επαληθευτές) είναι υλοποιημένος σε Java/Kotlin, με χρήση Android APIs και ασφαλούς αποθήκευσης (Secure Storage, Android Keystore). Οι λειτουργίες κρυπτογράφησης, υπογραφών και επεξεργασίας credentials υποστηρίζονται από ειδικά modules που εφαρμόζουν πρότυπα όπως SD-JWT, COSE και JWS, με υλοποίηση σε Java, ενώ τα OpenID-based flows διαχειρίζονται από ξεχωριστή βιβλιοθήκη OIDC client. Όλες οι ενότητες συντονίζονται μέσω κοινής αρχιτεκτονικής όπως περιγράφεται στο EUDI Wallet Architecture and Reference Framework (EUDIW - ARF), διασφαλίζοντας τη συνοχή, την ασφάλεια και τη συμμόρφωση με τις ευρωπαϊκές προδιαγραφές (EUDIW GitHub - Repositories, 2025; European Commission, 2025).

Τεχνικά Χαρακτηριστικά:

- Διαχείριση Διαπιστευτηρίων: Επιτρέπει στους χρήστες να λαμβάνουν, βλέπουν, αποθηκεύουν και αφαιρούν VCs με έλεγχο μέσω PIN ή βιομετρικών (π.χ. Android KeyStore, Secure Enclave). (EUDI – GitHub, 2025; Zahra Ebadi Ansaroudi et al, 2025)
- Αποθήκευση Κρυπτογραφικών Κλειδιών στη Συσκευή: Τα κρυπτογραφικά κλειδιά αποθηκεύονται σε τοπική, ασφαλή τοποθεσία (TEE/KeyStore/SE) χωρίς αποστολή σε servers. (Zahra Ebadi Ansaroudi et al, 2025)
- Επιλεκτική Διαμοίραση και Ιδιωτικότητα: Υποστηρίζει επέκταση σε ISO mDL και SD-JWT με δυνατότητα εκλεκτικής αποκάλυψης στοιχείων και unlinkability. (EUDI Wallet - ARF, 2025)
- Ροές Παρουσίασης Διαπιστευτηρίων: Υποστηρίζει τόσο proximity (QR/ISO mDL) όσο και remote OpenID4VP flows, με ενσωμάτωση Presentation Definition & DCQL. (EUDI – GitHub, 2025)

3.3.3 Επαληθευτής (Verifier)

Ο Verifier είναι το σύστημα που αναλαμβάνει την επαλήθευση των Verifiable Presentations που παρέχει το Wallet. Αποτελεί βασικό τεχνολογικό κρίκο στο οικοσύστημα EUDIW, καθώς παρέχει την υποδομή για την επαλήθευση ταυτότητας, ιδιοτήτων ή πιστοποιήσεων ενός χρήστη, σύμφωνα με τα αιτήματα παρουσίασης και τους κανόνες του OpenID for Verifiable Presentations (OpenID4VP).

Αναπτύσσεται κυρίως ως backend service σε Kotlin, και αξιοποιεί RESTful APIs για την αλληλεπίδραση με το Wallet. Εφαρμόζει OpenID4VP authorization ροές για ασφαλή request & response, αξιοποιώντας presentation policies όπως τα Presentation Definitions (PD) και τα ερωτήματα DCQL. Ο Verifier δεν επικοινωνεί ποτέ απευθείας με τον Issuer, διατηρώντας έτσι την αρχή της ιδιωτικότητας (privacy-preserving model). (EUDI - ARF, 2025; EUDI - GitHub, 2025)

Τεχνικά Χαρακτηριστικά:

- OpenID4VP Authorization Requests: Ο Verifier δημιουργεί authorization request που περιλαμβάνει presentation definition ή DCQL query, και τα αποστέλλει στο Wallet μέσω QR code ή redirect URI.
- Κρυπτογραφική Επαλήθευση: Επαληθεύει τις υπογραφές των credentials μέσω JWKs των issuers, ελέγχει expiry, proof of possession και consistency.
- Ενσωμάτωση Trust-Framework: Υποστηρίζει έλεγχο εγκυρότητας issuer μέσω trust registries, metadata services και validation policies.
- Ο Verifier μπορεί να έχει frontend διεπαφή χρήστη για real-time έλεγχο (π.χ. web interface), ή να λειτουργεί ως server-side service ενσωματωμένο σε τρίτες εφαρμογές. Στο πλαίσιο του EUDIW, έχουν παρασχεθεί παραδείγματα reference implementations για επαλήθευση διαπιστευτηρίων όπως ταυτότητες και άδειες κυκλοφορίας, μέσω web UI.

(EUDI – GitHub, 2025; European Commission, 2025)

Κεφάλαιο 4^ο

4 Υλοποίηση – Υποστήριξη Ελληνικής Ακαδημαϊκής Ταυτότητας στο Οικοσύστημα του EUDI

Η πρακτική υλοποίηση της παρούσας πτυχιακής εργασίας επικεντρώνεται στη λειτουργική ενσωμάτωση της Ελληνικής Ακαδημαϊκής Ταυτότητας στο οικοσύστημα της Ευρωπαϊκής Ψηφιακής Ταυτότητας (EUDI), αξιοποιώντας το διαθέσιμο ανοικτό κώδικα που προσφέρει η Ευρωπαϊκή Επιτροπή. Στόχος ήταν η ανάπτυξη ενός πλήρους σεναρίου που να καλύπτει τη ροή ενός Verifiable Credential, από την έκδοσή και τη διαχείρισή του από τον χρήστη μέσω Wallet, μέχρι την παρουσίαση και επαλήθευσή του από κάποιον Verifier.

Η επιλογή της Ελληνικής Ακαδημαϊκής Ταυτότητας ως σενάριο χρήσης δεν είναι τυχαία. Αφορά ένα μέσο ταυτοποίησης ευρείας χρήσης για τους φοιτητές, το οποίο φέρει στοιχεία όπως το ονοματεπώνυμο φοιτητή, τον αριθμό μητρώου, το ακαδημαϊκό ίδρυμα και το έτος σπουδών. Η ψηφιοποίηση αυτού του στοιχείου με τη μορφή επαληθεύσιμου ψηφιακού διαπιστευτηρίου καθιστά εφικτή τη χρήση του σε διαδικτυακά περιβάλλοντα ταυτοποίησης, τόσο εντός όσο και εκτός Ελλάδος.

Το σενάριο που υλοποιήθηκε περιλαμβάνει τα ακόλουθα γενικά βήματα:

- Τροποποίηση και παραμετροποίηση του Issuer, έτσι ώστε να μπορεί να εκδώσει την Ελληνική ακαδημαϊκή ταυτότητα με βάση τις απαιτήσεις παρουσίασης του EUDIW.
- Διαμόρφωση reverse proxy και προσαρμογή του Wallet, ώστε να μπορεί να επικοινωνεί με τον νέο Issuer και να αποθηκεύει την ταυτότητα.
- Δημιουργία custom Verifier web interface, το οποίο να μπορεί να ζητήσει και να επαληθεύσει το συγκεκριμένο credential με χρήση Presentation Definition.

Κατά την υλοποίηση αξιοποιήθηκε το cloud based αποθετήριο κώδικα GitHub για την αποθήκευση και τον έλεγχο εκδόσεων (version control) του κώδικα. Η εργασία οργανώθηκε σε διακριτά αποθετήρια (repositories), ώστε κάθε στοιχείο (Issuer, Wallet, Verifier) να μπορεί να τροποποιηθεί και να αναπτυχθεί ανεξάρτητα. Τα αποθετήρια των Issuer και Wallet αποτελούν κλώνους των επίσημων project της Ε.Ε., των οποίων η λειτουργία επεκτάθηκε, ενώ το αποθετήριο του Verifier περιέχει κώδικα ανεπτυγμένο εκ του μηδενός. Τα αποθετήρια αυτά βρίσκονται συγκεντρωμένα στο εξής κεντρικό αποθετήριο: <https://github.com/konsklav/gracid-eudiw-thesis>

Η υλοποίηση έχει γίνει αποκλειστικά με χρήση τεχνολογιών ανοιχτού κώδικα, και αποτελεί ρεαλιστικό σενάριο ενσωμάτωσης ενός Εθνικού διαπιστευτηρίου στο Ευρωπαϊκό οικοσύστημα ταυτοτήτων.

4.1 Τροποποίηση και παραμετροποίηση του Εκδότη (Issuer) για έκδοση Ελληνικής Ακαδημαϊκής Ταυτότητας

4.1.1 Αναλυτική Περιγραφή

Με σκοπό την επέκταση του υπάρχοντος Issuing Service του EUDI οικοσυστήματος, προστέθηκε η δομή της ακαδημαϊκής ταυτότητας στα υποστηριζόμενα διαπιστευτήρια και τροποποιήθηκαν οι ρυθμίσεις (configurations) του.

Η δήλωση του νέου διαπιστευτηρίου, έγινε σε μορφή JSON, βασισμένη στο πρότυπο του mDL (κινητή άδεια οδήγησης). Το JSON αρχείο αποτελεί την δήλωση ενός νέου τύπου διαπιστευτηρίου και περιέχει τα doctype και scope (αναγνωριστικά του διαπιστευτηρίου), τα υποστηριζόμενα bindings και proof types, την μορφή έκδοσης του (mso_mdoc - πρότυπο υπογραφής/ασφάλισης mobile documents), τον τρόπο αναπαράστασής του στο wallet και τα πεδία δεδομένων της ταυτότητας (claims).

```
1 {
2   "eu.europa.ec.eudi.greek_academic_id_mdoc": {
3     "format": "mso_mdoc",
4     "doctype": "eu.europa.ec.eudi.greek_academic_id.1",
5     "scope": "eu.europa.ec.eudi.greek_academic_id.1",
6     "cryptographic_binding_methods_supported": [
7       "jwk", "cose_key"
8     ],
9     "credential_signing_alg_values_supported": [
10      "ES256"
11    ],
12    "proof_types_supported": {
13      "jwt": {
14        "proof_signing_alg_values_supported": [
15          "ES256"
16        ]
17      },
18      "cwt": {
19        "proof_signing_alg_values_supported": [
20          "ES256"
21        ],
22        "proof_alg_values_supported": [
23          "-7"
24        ],
25        "proof_crv_values_supported": [
26          1
27        ]
28      }
29    },
30    "display": [
31      {
32        "name": "Greek Academic ID",
33        "locale": "en",
34        "logo": {
35          "url": "https://examplestate.com/public/pid.png",
36          "alt_text": "A square figure of an Academic ID"
37        }
38      }
39    ]
40  }
41 }
```

Εικόνα 4.1.1 – Κώδικας JSON που περιέχει τη δομή της ακαδημαϊκής ταυτότητας.

Στην ενότητα claims περιγράφονται όλα τα πεδία που περιλαμβάνει το διαπιστευτήριο. Αφορά τόσο τα ίδια τα δεδομένα, όπως στοιχεία φοιτητή και ημερομηνίες, όσο και τα μεταδεδομένα που τα συνοδεύουν για λειτουργικούς σκοπούς, όπως το αν αποτελούν υποχρεωτικά πεδία, το αν πηγάζουν από τον χρήστη ή τον εκδότη και ο τύπος τους.

```
40 "claims": {
41   "eu.europa.ec.eudi.greek_academic_id.1": {
42     "given_name": {
43       "mandatory": true,
44       "value_type": "string",
45       "source": "user",
46       "display": [
47         {
48           "name": "Given Name",
49           "locale": "en"
50         }
51       ],
52       "family_name": {
53         "mandatory": true,
54         "value_type": "string",
55         "source": "user",
56         "display": [
57           {
58             "name": "Family Name",
59             "locale": "en"
60           }
61         ],
62         "given_name_gr": {
63           "mandatory": true,
64           "value_type": "string",
65           "source": "user",
66           "display": [
67             {
68               "name": "Given Name in Greek",
69               "locale": "en"
70             }
71           ],
72           "family_name_gr": {
73             "mandatory": true,
74             "value_type": "string",
75             "source": "user",
76             "display": [
77               {
78                 "name": "Family Name in Greek",
79                 "locale": "en"
80               }
81             ]
82           }
83         }
84       ]
85     }
86   }
87 }
```

Εικόνα 4.1.2 – Κώδικας JSON που περιέχει τη δομή της ακαδημαϊκής ταυτότητας (συνέχεια – δομή των claims).

Στην συνέχεια, τροποποιήθηκαν τα configuration αρχεία με σκοπό την αναγνώριση του νέου διαπιστευτηρίου (για έκδοση όπως τα υπάρχοντα διαπιστευτήρια), την δήλωση της προσωπικής διεύθυνσης του υπολογιστή (localhost) ως νέο base URL της υπηρεσίας, καθώς και τη δήλωση ενός έμπιστου path στον υπολογιστή, στο οποίο περιέχονται τα κρυπτογραφημένα κλειδιά των απαραίτητων, για την έκδοση, πιστοποιητικών.

Τα συγκεκριμένα πιστοποιητικά ονομάζονται IACA (Issuer Authority Certificate Authority). Χρησιμοποιούνται για την επικύρωση της αυθεντικότητας του εκδότη στο πλαίσιο των mobile documents (mdoc) και είναι κρίσιμα για να μπορεί το Wallet να εμπιστευτεί το διαπιστευτήριο.

```
68 "supported_credentials": [
69   "eu.europa.ec.eudi.greek_academic_id_mdoc",
```

Εικόνα 4.1.3 – Τμήμα κώδικα των configuration αρχείων σε Python για προσθήκη της ακαδημαϊκής ταυτότητας στα υποστηριζόμενα διαπιστευτήρια.


```
87     # Academic ID namespace
88     ac_id_namespace = "eu.europa.ec.eudi.greek_academic_id.1"
89
90     # Academic ID validity in days
91     ac_id_validity = 90
92
93     # Academic ID issuing Authority
94     ac_id_issuing_authority = "Test Academic ID issuer"
```

Εικόνα 4.1.4 – Τμήμα κώδικα των configuration αρχείων σε Python για configuration του doctype της ακαδημαϊκής ταυτότητας.

```
267     config_doctype = {
268         "eu.europa.ec.eudi.greek_academic_id.1": {
269             "issuing_authority": ac_id_issuing_authority,
270             "organization_id": pid_organization_id,
271             "validity": ac_id_validity,
272             "organization_name": ac_id_issuing_authority,
273             "namespace": ac_id_namespace,
274         },
```

Εικόνα 4.1.5 – Τμήμα κώδικα των configuration αρχείων σε Python για configuration του doctype της ακαδημαϊκής ταυτότητας.

```
38     # service_url = "https://127.0.0.1:5000/"
39     #service_url = "https://localhost:5000/"
40     service_url = "https://similarly-up-raven.ngrok-free.app/"
```

Εικόνα 4.1.6 – Τμήμα κώδικα των configuration αρχείων σε Python για δήλωση custom διεύθυνσης της υπηρεσίας.

```
48     trusted_CAs_path = "/etc/eudiw/pid-issuer/cert/"
49
```

Εικόνα 4.1.7 – Τμήμα κώδικα των configuration αρχείων σε Python για δήλωση έμπιστου path τοπικής αποθήκευσης πιστοποιητικών.

4.1.2 Τεχνολογίες και Εργαλεία που Χρησιμοποιήθηκαν

- **Εργαλεία Ανάπτυξης:**
Η ανάπτυξη έγινε μέσω του Visual Studio Code, σε συνδυασμό με Git/GitHub για κλωνοποίηση του υπάρχοντος κώδικα, έλεγχο έκδοσης και παρακολούθηση αλλαγών.
- **Γλώσσα Προγραμματισμού:**
Ο κώδικας είναι υλοποιημένος στην γλώσσα προγραμματισμού Python, συγκεκριμένα στην έκδοση 3.9 και η επέκταση του πραγματοποιήθηκε στην έκδοση 3.10. Επιπλέον, έγινε χρήση JSON ως data-interchange format για την δήλωση του διαπιστευτηρίου.

- Web Framework:
Ο Issuer υλοποιείται με χρήση Flask, ενός ελαφριού web framework για Python, κατάλληλου για RESTful APIs και εύκολη επέκταση/παραμετροποίηση. Μέσω του framework αυτού εκτελέστηκε τοπικά η υπηρεσία.

4.1.3 Τρόπος Εγκατάστασης και Λειτουργίας

Για την εγκατάσταση και λειτουργία της εκτεταμένης μορφής του Issuer απαιτούνται τα εξής βήματα:

- Εγκατάσταση Python v. 3.9 ή 3.10
- Εγκατάσταση Flask v. 2.3 ή νεότερη
- Κλωνοποίηση του αποθετηρίου του Issuer (main branch):

```
“git clone https://github.com/konsklav/eudi-srv-web-issuing-eudiw-py.git”
```

- Μετάβαση στον φάκελο του έργου και δημιουργία εικονικού περιβάλλοντος (virtual environment):

```
“cd eudi-srv-web-issuing-eudiw-py  
  
python3 -m venv .venv”
```

- Εκκίνηση του εικονικού περιβάλλοντος:

```
“. .venv/bin/activate” (Linux/macOS)
```

```
“. .venv\Scripts\activate” (Windows)
```

- Ενημέρωση/Εγκατάσταση pip (αν χρειάζεται):

```
“python -m pip install --upgrade pip”
```

- Εγκατάσταση των απαραίτητων πακέτων (Flask, κλπ.):

```
“pip install -r app/requirements.txt”
```

- Ρύθμιση του FLASK_SECRET_KEY:

1. Αντιγραφή των αρχείων στο τερματικό: “cp app/app_config/__config_secrets.py app/app_config/config_secrets.py”
2. Επεξεργασία του αρχείου config_secrets.py:

```
Αλλαγή της γραμμής: «flask_secret_key = "secret_here"», σε:  
«flask_secret_key = os.getenv("FLASK_SECRET_KEY")»
```

3. Ορισμός μυστικού κλειδιού:

“export FLASK_SECRET_KEY="κάποιο_πολύ_ισχυρό_μυστικό”
(Linux/macOS - Terminal)

“set FLASK_SECRET_KEY=κάποιο_πολύ_ισχυρό_μυστικό”
(Windows - CMD)

■ Αποθήκευση πιστοποιητικών:

1. Αποθήκευση του εξής πιστοποιητικού:

https://github.com/konsklav/eudi-srv-web-issuing-eudiw-py/blob/main/api_docs/test_tokens/IACA-token/PIDIssuerCAUT01.pem.gz, σε PEM format, στο path:

“etc/eudiw/pid-issuer/cert”

2. Αποθήκευση του εξής πιστοποιητικού:

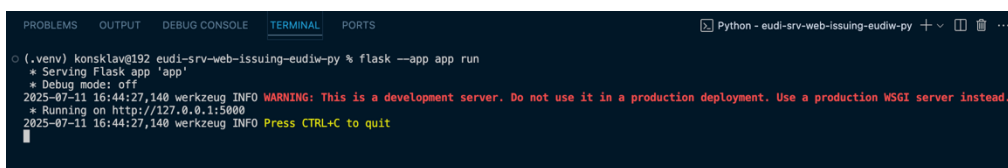
https://github.com/konsklav/eudi-srv-web-issuing-eudiw-py/blob/main/api_docs/test_tokens/DS-token/PID-DS-0002.zip, σε DER format, στο path: “etc/eudiw/pid-issuer/cert” και του decrypted κλειδί, σε PEM format, στο path: “etc/eudiw/pid-issuer/privKey”.

3. Για εξαγωγή του decrypted κλειδιού από το encrypted εκτελείται στο τερματικό το εξής: “openssl ec -in PID-DS-0002.pid-ds-0002.key.pem -out PID-DS-0002-decrypted.key.pem”

■ Τοπική εκτέλεση του service (μέσω του εικονικού περιβάλλοντος):

“flask --app app run”

(τρέχει στο <http://127.0.0.1:5000/> ή <http://localhost:5000/>)



Εικόνα 4.1.8 – Στιγμιότυπο εκτέλεσης της υπηρεσίας τοπικά με χρήση Flask.

4.1.4 Παράδειγμα Εκτέλεσης

Παρουσιάζεται στην ενότητα 4.2.4, σε συνδυασμό με την εκτέλεση της Android εφαρμογής του Wallet.

4.2 Διαμόρφωση Reverse Proxy και προσαρμογή του Πορτοφολιού (Wallet) για έκδοση Ελληνικής Ακαδημαϊκής Ταυτότητας με χρήση του εκτεταμένου Εκδότη (Issuer)

4.2.1 Αναλυτική Περιγραφή

Μετά την επέκταση του Issuer για έκδοση της Ακαδημαϊκής Ταυτότητας, το επόμενο βήμα ήταν η διασύνδεσή του με την Android εφαρμογή του EUDI Wallet, ώστε να μπορεί ο χρήστης να παραλάβει και να αποθηκεύσει το διαπιστευτήριο στη συσκευή του.

Για τον σκοπό αυτό, χρησιμοποιήθηκε το ngrok, ένα εργαλείο που δημιουργεί έναν reverse proxy προς τη διεύθυνση localhost και την εκθέτει προσωρινά στο διαδίκτυο μέσω ενός δημόσιου domain (π.χ. <https://xyz.ngrok.app>). Ο reverse proxy επιτρέπει την ανακατεύθυνση αιτημάτων από εξωτερικούς clients, όπως το Wallet, προς το τοπικό περιβάλλον ανάπτυξης στο οποίο εκτελείται ένα πρόγραμμα. Έτσι, ο Issuer κατέστη προσβάσιμος εκτός του τοπικού δικτύου για τις ανάγκες διασύνδεσης με το πορτοφόλι.

Στην πράξη, αποκτήθηκε αρχικά μία μοναδική στατική διεύθυνση (domain) μέσω του ngrok (<https://similarly-up-raven.ngrok-free.app>), εγκαταστάθηκε το λογισμικό του ngrok και στην συνέχεια εκτελέστηκε η εξής εντολή στο τερματικό:

```
“ngrok http --domain=similarly-up-raven.ngrok-free.app 127.0.0.1:5000”
```

, η οποία εκκινεί την διαδικασία κατοπτρισμού της localhost διεύθυνσης (στην οποία εκτελείται ο Issuer) στο ενεργό στατικό domain του ngrok.



```
ngrok (Ctrl+C to quit)
Make HTTP calls to internal services from your gateway (dev preview): https://ngrok.com/r/http-request

Session Status      online
Account             konsklav (Plan: Free)
Version             3.23.3
Region              Europe (eu)
Latency              51ms
Web Interface        http://127.0.0.1:4040
Forwarding            https://similarly-up-raven.ngrok-free.app -> http://127.0.0.1:5000

Connections          ttl    opn    rt1    rt5    p50    p90
                     0      0      0.00   0.00   0.00   0.00
```

Εικόνα 4.2.1 – Στιγμιότυπο λειτουργίας του reverse proxy κατοπτρισμού της διεύθυνσης localhost στο στατικό domain του ngrok.

Τέλος, για να είναι εφικτή η πρόσβαση του Wallet σε αυτή την διεύθυνση με σκοπό την επικοινωνία με τον Issuer κατά τη ροή έκδοσης, τροποποιήθηκε το URL του Issuer μέσα στον πηγαίο κώδικα του Wallet, ώστε να δείχνει στο δημόσιο domain που παρείχε το ngrok αντί για την προκαθορισμένη τιμή.

4.2.2 Τεχνολογίες και Εργαλεία που Χρησιμοποιήθηκαν

- Εργαλεία Ανάπτυξης και Εκτέλεσης:
Η επέκταση του κώδικα και η εκτέλεση της εφαρμογής έγινε μέσω του περιβάλλοντος του Android Studio. Για την κλωνοποίηση του υπάρχοντος κώδικα, έλεγχο έκδοσης και παρακολούθηση αλλαγών χρησιμοποιήθηκαν τα Git/GitHub. Τέλος, για τις δοκιμές έγινε χρήση της υπηρεσίας διαχείρισης API: ngrok.
- Γλώσσα Προγραμματισμού και Framework:
Ο κώδικας είναι υλοποιημένος στην σύγχρονη γλώσσα ανάπτυξης Android εφαρμογών, Kotlin. Το project βασίζεται στο Android SDK, με χρήση Jetpack libraries για τον χειρισμό διεπαφής και flows.

4.2.3 Τρόπος Εγκατάστασης και Λειτουργίας

Για την εγκατάσταση και λειτουργία της εφαρμογής για έκδοση Ακαδημαϊκής Ταυτότητας απαιτούνται τα εξής βήματα:

- Εγκατάσταση Android Studio
- Κλωνοποίηση του αποθετηρίου του Wallet (main branch):

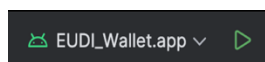
“git clone <https://github.com/konsklav/eudi-app-android-wallet-ui.git>”

- Απόκτηση στατικού domain μέσω του: <https://ngrok.com>
- Εγκατάσταση του ngrok όπως περιγράφεται στο: <https://ngrok.com>
- Εκτέλεση, στο τερματικό, της εντολής:

“ngrok http --domain=το-στατικό-σας-domain 127.0.0.1:5000”

(βασικό προαπαιτούμενο: η τοπική εκτέλεση του Issuer στην διεύθυνση localhost, όπως παρουσιάζεται στην ενότητα 4.1.3)

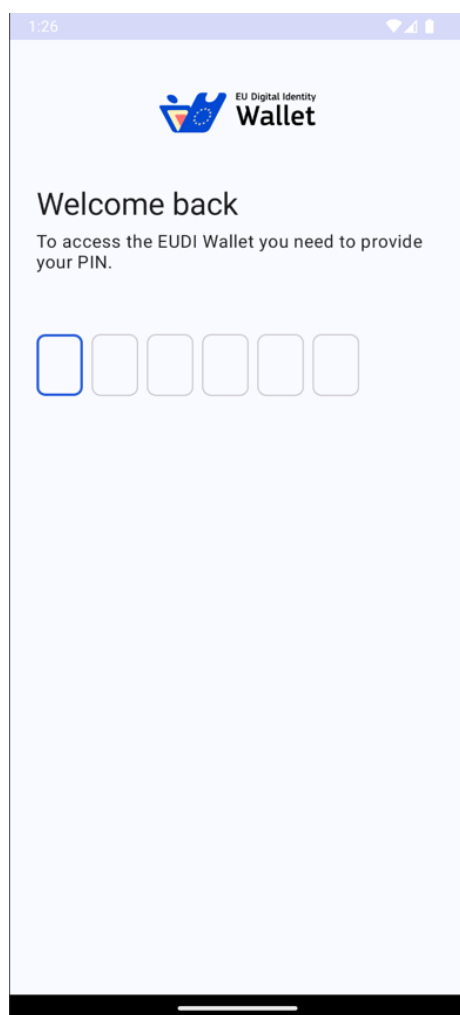
- Εκτέλεση της εφαρμογής μέσω του Android Studio, πατώντας το κουμπί εκτέλεσης:



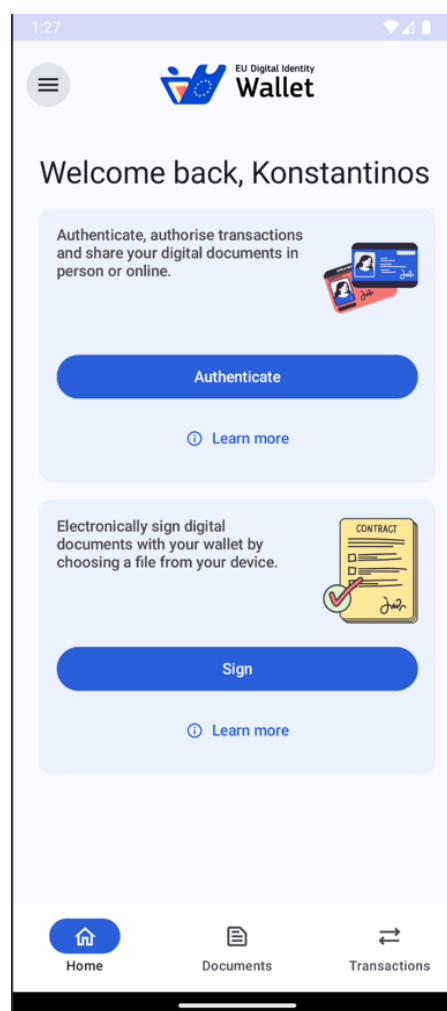
Εικόνα 4.2.2 – Κουμπί εκτέλεσης εφαρμογής Wallet στο Android Studio.

4.2.4 Παράδειγμα Εκτέλεσης

Κατά την έναρξη της εφαρμογής ζητείται η υποβολή του κωδικού PIN του χρήστη. Αυτός ο κωδικός είναι προσωπικός, αρχικοποιείται κατά την πρώτη φορά χρήσης της εφαρμογής και χρησιμοποιείται τόσο για την ασφαλή είσοδο όσο και για την αδειοδότηση κοινοποίησης των διαπιστευτηρίων σε κάποιον επαληθευτή. Μετά την εισαγωγή του, εμφανίζεται η κεντρική σελίδα, στην οποία ο χρήστης έχει την επιλογή να πραγματοποιήσει κάποια αυθεντικοποίηση ή να μεταβεί στα ψηφιακά του έγγραφα (Documents).

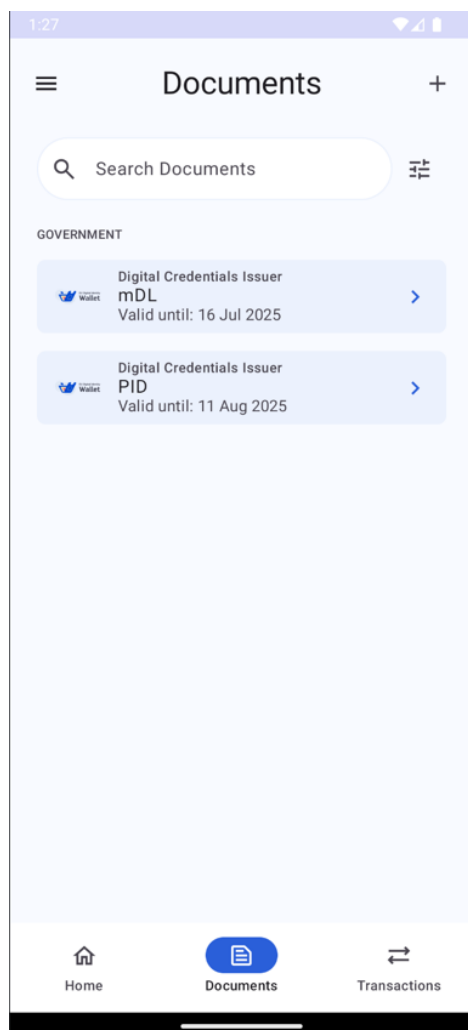


Εικόνα 4.2.3 – Οθόνη εισόδου στο Wallet.

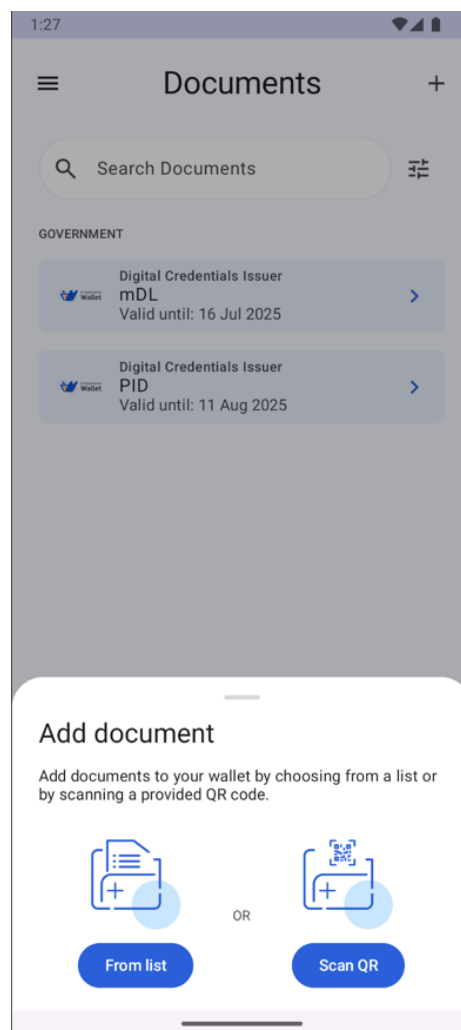


Εικόνα 4.2.4 – Αρχική οθόνη του Wallet.

Η περίπτωση της αυθεντικοποίησης παρουσιάζεται στην ενότητα 4.3.4 . Σε αυτό το παράδειγμα παρουσιάζεται η περίπτωση έκδοσης μίας ψηφιακής ακαδημαϊκής ταυτότητας και για το σκοπό αυτό πραγματοποιείται μετάβαση στην σελίδα των εγγράφων (Documents). Από εκεί επιλέγεται το κουμπί “+”, για εμφάνιση των τρόπων έκδοσης.

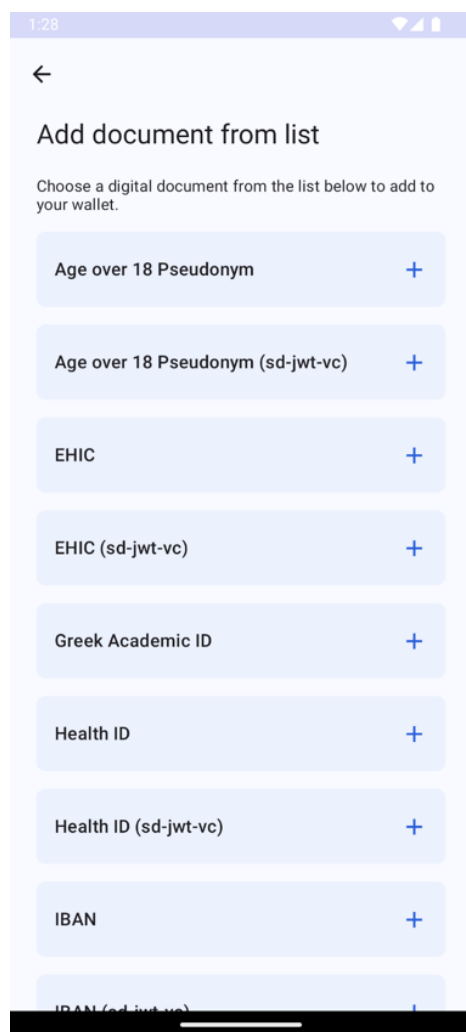


Εικόνα 4.2.5 – Οθόνη εγγράφων του Wallet.

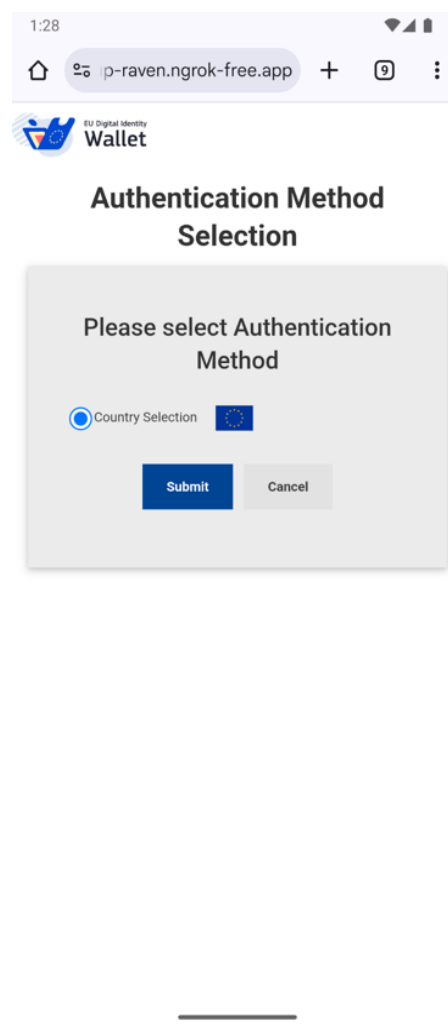


Εικόνα 4.2.6 – Οθόνη επιλογής τρόπου προσθήκης εγγράφων στο Wallet.

Επιλέγεται η επιλογή έκδοσης από τη λίστα των διαθέσιμων, από τον εκδότη (Issuer), διαπιστευτηρίων. Στη λίστα είναι εμφανής η επιλογή της ακαδημαϊκής ταυτότητας (Greek Academic Id), που δηλώνει επιτυχή σύνδεση με τον «νέο» Issuer. Επιλέγοντας την γίνεται ανακατεύθυνση στην σελίδα του Issuer και εμφανίζονται οι επιλογές τρόπου έκδοσης. Σε αυτό το παράδειγμα υπάρχει η επιλογή χώρας.

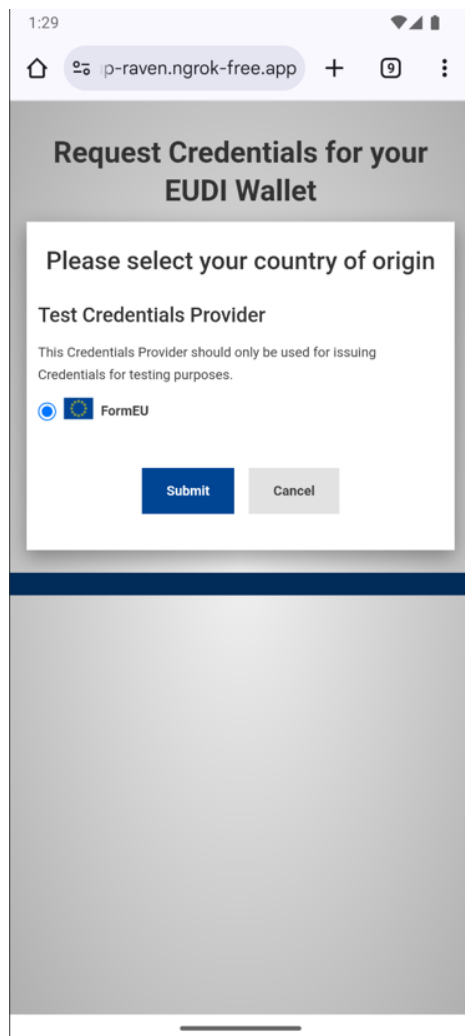


Εικόνα 4.2.7 – Λίστα διαθέσιμων πιστοποιητικών.



Εικόνα 4.2.8 – Σελίδα επιλογής μεθόδου αυθεντικοποίησης.

Μετά την επιλογή αυτή εμφανίζονται οι διαθέσιμες χώρες που εκδίδουν το συγκεκριμένο πιστοποιητικό. Για τον σκοπό του παραδείγματος, λόγω κατάστασης ανάπτυξης και δοκιμής, χρησιμοποιείται η φόρμα της Ε.Ε.



1:29


ip-raven.ngrok-free.app

Request Credentials for your EUDI Wallet

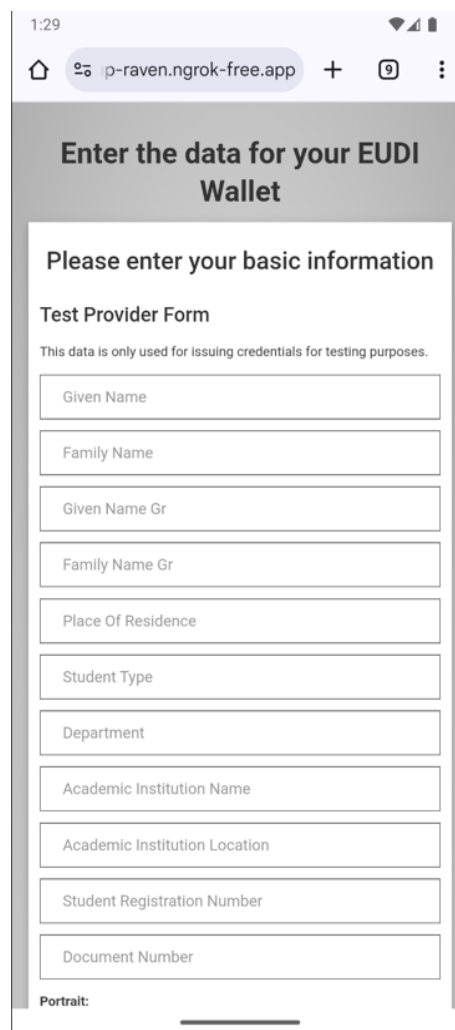
Please select your country of origin

Test Credentials Provider

This Credentials Provider should only be used for issuing Credentials for testing purposes.

☒  FormEU

Εικόνα 4.2.9 – Σελίδα επιλογής χώρας προέλευσης.



1:29

ip-raven.ngrok-free.app

Enter the data for your EUDI Wallet

Please enter your basic information

Test Provider Form

This data is only used for issuing credentials for testing purposes.

Given Name

Family Name

Given Name Gr

Family Name Gr

Place Of Residence

Student Type

Department

Academic Institution Name

Academic Institution Location

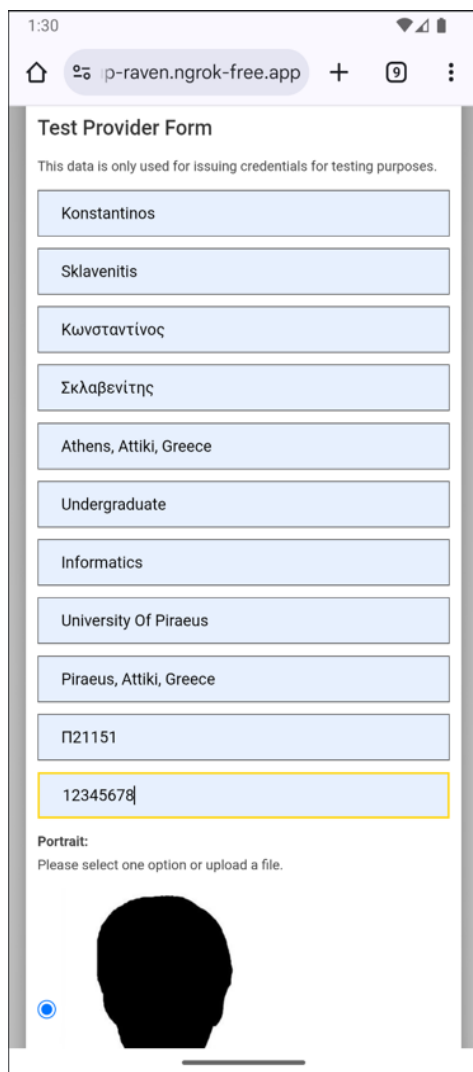
Student Registration Number

Document Number

Portrait:

Εικόνα 4.2.10 – Φόρμα εκχώρησης των δεδομένων του διαπιστευτηρίου.

Συμπληρώνεται και υποβάλλεται η φόρμα με τα στοιχεία του φοιτητή.



1:30

ip-raven.ngrok-free.app

Test Provider Form

This data is only used for issuing credentials for testing purposes.

Konstantinos

Sklaenitis

Κωνσταντίνος

Σκλαβενίτης

Athens, Attiki, Greece

Undergraduate

Informatics

University Of Piraeus


Piraeus, Attiki, Greece

Π21151

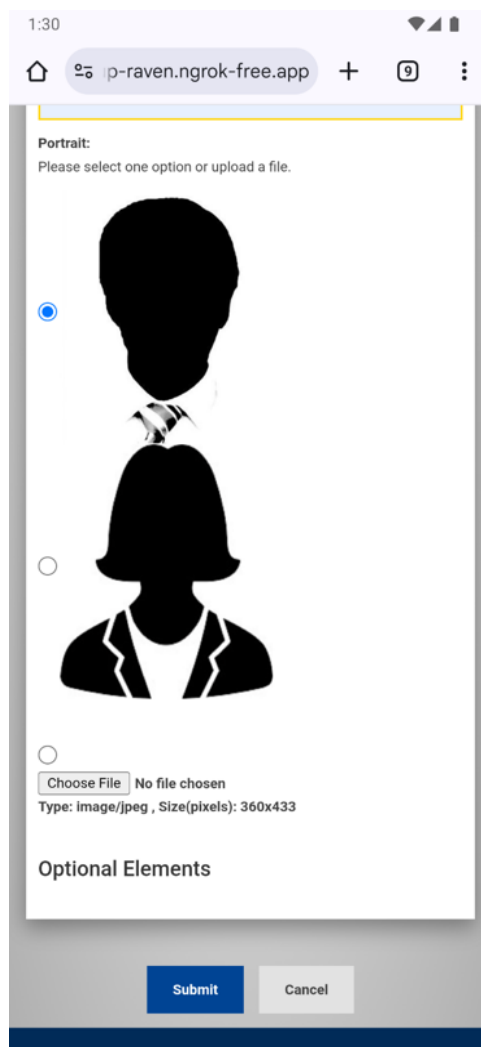
12345678

Portrait:

Please select one option or upload a file.

☒ 

Εικόνα 4.2.11 – Συμπληρωμένη φόρμα.





1:30

ip-raven.ngrok-free.app

Portrait:

Please select one option or upload a file.

☒ 

☐ 

☐ No file chosen
Type: image/jpeg , Size(pixels): 360x433

Optional Elements

Εικόνα 4.2.12 – Συμπληρωμένη φόρμα με επιλογή υποβολής.

Στη συνέχεια απαιτείται έλεγχος ορθότητας των στοιχείων και πραγματοποιείται οριστική υποβολή.

1:31

ip-raven.ngrok-free.app

Authorize data from your EUDI Wallet

Please confirm your information

Provider Form Authentication

Please confirm if your data is right

Greek Academic ID

given_name
Konstantinos

family_name
Sklaivenitis

given_name_gr
Κωνσταντίνος

family_name_gr
Σκλαβενίτης

place_of_residence
Athens, Attiki, Greece

student_type
Undergraduate

department
Informatics

academic_institution_name

Εικόνα 4.2.13 – Σελίδα ελέγχου των πληροφοριών.

1:31

ip-raven.ngrok-free.app

student_registration_number
Π21151

document_number
12345678

portrait

estimated_issuance_date
2025-07-12

estimated_expiry_date
2025-10-10

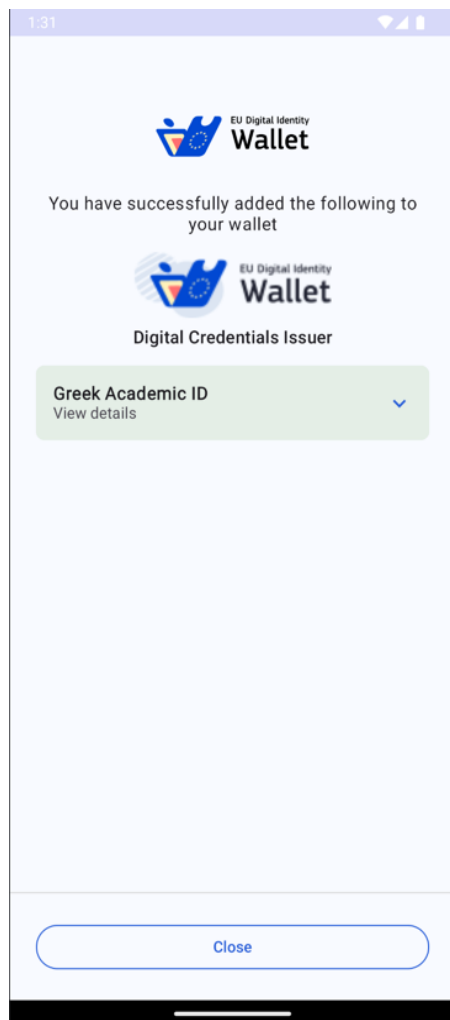
issuing_country
FC

issuing_authority
Test Academic ID issuer

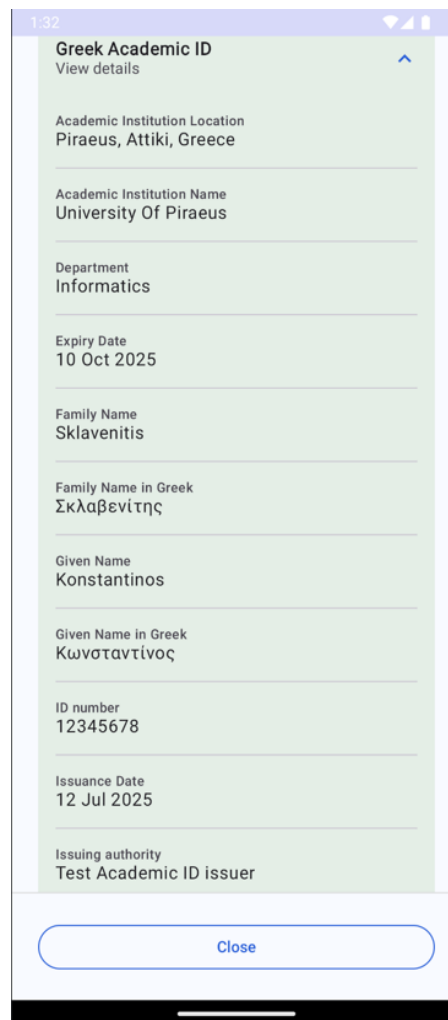
Authorize Cancel

Εικόνα 4.2.14 – Σελίδα ελέγχου των πληροφοριών (συνέχεια) με επιλογή εξουσιοδότησης.

Με την οριστική υποβολή πραγματοποιείται εκ νέου ανακατεύθυνση στην εφαρμογή του Wallet, όπου και προβάλλεται μήνυμα επιτυχούς έκδοσης του διαπιστευτηρίου και υπάρχει η δυνατότητα προβολής των στοιχείων του.

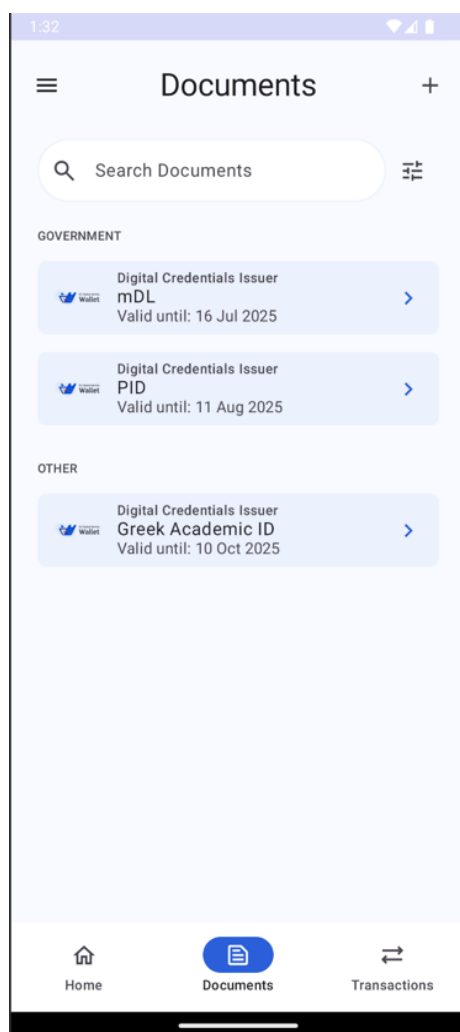


Εικόνα 4.2.15 – Οθόνη επιτυχούς έκδοσης διαπιστευτηρίου.

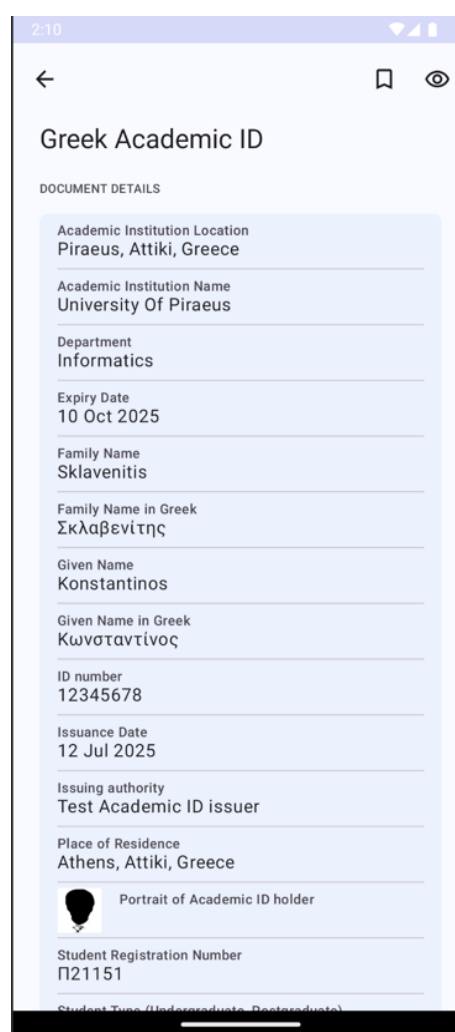


Εικόνα 4.2.16 – Οθόνη στοιχείων νέου διαπιστευτηρίου.

Τέλος, επιλέγοντας κλείσιμο (Close), γίνεται ανακατεύθυνση στα έγγραφα του χρήστη, τα οποία πλέον περιλαμβάνουν επιτυχώς και την ακαδημαϊκή ταυτότητα.



Εικόνα 4.2.17 – Νέα οθόνη εγγράφων του Wallet.



Εικόνα 4.2.18 – Στοιχεία νέου διαπιστευτηρίου.

4.3 Υλοποίηση Επαλήθευτή (Verifier) Ελληνικών Ακαδημαϊκών Ταυτοτήτων

4.3.1 Αναλυτική Περιγραφή

Για να ολοκληρωθεί ο κύκλος ζωής του διαπιστευτηρίου, αναπτύχθηκε ένα πλήρως λειτουργικό περιβάλλον επαλήθευσης της Ελληνικής Ακαδημαϊκής Ταυτότητας, σύμφωνα με τα πρότυπα του EUDI. Το περιβάλλον αυτό αποτελεί μία προσαρμοσμένη frontend διεπαφή χρήστη, η οποία χρησιμοποιεί το online backend σύστημα επαλήθευσης του EUDI οικοσυστήματος.

Πιο συγκεκριμένα, αξιοποιούνται τα υπάρχοντα backend endpoints που είναι ενεργά στη διεύθυνση: <https://dev.verifier-backend.eudiw.dev/>, η οποία αντικατοπτρίζει το σύστημα που βρίσκεται στο επίσημο αποθετήριο του EUDI (<https://github.com/eu-digital-identity-wallet/eudi-srv-web-verifier-endpoint-23220-4-kt>), υλοποιημένο σε Kotlin.

Η εφαρμογή έχει προσαρμοστεί με τρόπο τέτοιο, ώστε να αφορά επαλήθευση μόνο της ακαδημαϊκής ταυτότητας. Για τον σκοπό αυτό, περιέχει μία έτοιμη δομή ενός presentation definition (PD) με τα ζητούμενα πεδία της ακαδημαϊκής ταυτότητας προς επαλήθευση.

```
1  {
2    "id": "verify-greek-academic-id",
3    "input_descriptors": [
4      {
5        "id": "eu.europa.ec.eudi.greek_academic_id.1",
6        "format": {
7          "mso_mdoc": {
8            "alg": [
9              "ES256"
10             ]
11          }
12        },
13        "constraints": {
14          "limit_disclosure": "preferred",
15          "fields": [
16            {
17              "path": [
18                "$['eu.europa.ec.eudi.greek_academic_id.1']['given_name']"
19              ],
20              "intent_to_retain": false
21            },
22            {
23              "path": [
24                "$['eu.europa.ec.eudi.greek_academic_id.1']['family_name']"
25              ],
26              "intent_to_retain": false
27            },
28            {
29              "path": [
30                "$['eu.europa.ec.eudi.greek_academic_id.1']['given_name_gr']"
31              ],
32              "intent_to_retain": false
33            },
34            {
35              "path": [
36                "$['eu.europa.ec.eudi.greek_academic_id.1']['family_name_gr']"
37              ],
38              "intent_to_retain": false
39            }
40          ]
41        }
42      }
43    ]
44  }
```

Εικόνα 4.3.1 – Presentation Definition για επαλήθευση της ακαδημαϊκής ταυτότητας σε μορφή JSON.

Αυτό το PD χρησιμοποιείται στο presentation request που αποστέλλεται στο verifier endpoint, κατά την αρχικοποίηση της συνεδρίας επαλήθευσης, συνοδευόμενο από τις υπόλοιπες επιλογές/ρυθμίσεις για τον τρόπο με τον οποίο επιθυμεί να εκκινήσει την επαλήθευση και να λάβει την απάντηση από το Wallet.

```
12 const response = await fetch('https://dev.verifier-backend.eudiw.dev/ui/presentations', {  
13   method: 'POST',  
14   headers: {  
15     'Content-Type': 'application/json'  
16   },  
17   body: JSON.stringify({ // Basic request body  
18     "type": "vp_token",  
19     "presentation_definition": presentationDefinition,  
20     "dcql_query": null,  
21     "nonce": nonce,  
22     "response_mode": "direct_post",  
23     "jar_mode": "by_reference",  
24     "request_uri_method": "post"  
25   })  
26 });
```

Εικόνα 4.3.2 – Τμήμα κώδικα σε JavaScript για την αρχικοποίηση της συνεδρίας επαλήθευσης με presentation request.

Μετά την αρχικοποίηση της συνεδρίας, η εφαρμογή λαμβάνει από το backend service την απάντηση με τα δεδομένα της συναλλαγής, τα οποία χρησιμοποιεί για την δημιουργία ενός QR-Code το οποίο και προβάλλει για σκανάρισμα από το Wallet.

```
35 // Create QR-Code Uri based on the response from the session initializer (rerenders everytime the state of the attributes used changes)  
36 const qrCodeUri = `eudi-openid4vp://?client_id=${encodeURIComponent(clientId)}  
37   &request_uri=${encodeURIComponent(requestUri)}  
38   &request_uri_method=${encodeURIComponent(requestUriMethod)}`;
```

Εικόνα 4.3.3 – Τμήμα κώδικα σε JavaScript για την δημιουργία του URI που περιγράφει το QR-Code.

Στη συνέχεια, αναμένει τις απαραίτητες διεργασίες από την εφαρμογή του Wallet, κρατώντας ανοικτό δίαυλο επικοινωνίας με το backend service εν αναμονή κάποιας απάντησης.

Μόλις λάβει την απάντηση (δεδομένου της αποδοχής επαλήθευσης από το wallet και της κοινοποίησης των στοιχείων), την επεξεργάζεται, εξάγει τα στοιχεία και τα προβάλλει παράλληλα με μήνυμα επιτυχούς επαλήθευσης.

4.3.2 Τεχνολογίες και Εργαλεία που Χρησιμοποιήθηκαν

- Εργαλεία Ανάπτυξης και Εκτέλεσης:

Η ανάπτυξη της εφαρμογής πραγματοποιήθηκε μέσω του Visual Studio Code, ενώ για την εγκατάσταση εξαρτήσεων και τη διαχείριση του περιβάλλοντος χρησιμοποιήθηκε το npm (Node Package Manager). Για τη δοκιμαστική εκτέλεση σε τοπικό περιβάλλον χρησιμοποιήθηκε το Vite, ένα σύγχρονο εργαλείο ανάπτυξης και build για frontend εφαρμογές. Παράλληλα, για τον έλεγχο εκδόσεων χρησιμοποιήθηκαν τα Git/GitHub.

- Γλώσσα Προγραμματισμού και Framework:

Η εφαρμογή είναι υλοποιημένη σε JavaScript (ES6), χρησιμοποιώντας το React framework για την ανάπτυξη του UI. Ο σχεδιασμός της εφαρμογής βασίζεται στη λογική των components, ενώ αξιοποιούνται επίσης custom React hooks για τον χειρισμό του polling, του session initialization και της επαλήθευσης των credentials.

- Βοηθητικά Εργαλεία και Ροές:

Για την επεξεργασία των Presentation Definitions χρησιμοποιούνται αρχεία τύπου JSON. Ενώ, για την δημιουργία του QR-Code χρησιμοποιείται η βιβλιοθήκη react-qrcode (QRCodeDisplay) του React.

4.3.3 Τρόπος Εγκατάστασης και Λειτουργίας

- Εγκατάσταση Node.js v.18 ή νεότερη
- Εγκατάσταση των εξαρτήσεων:

“npm install”

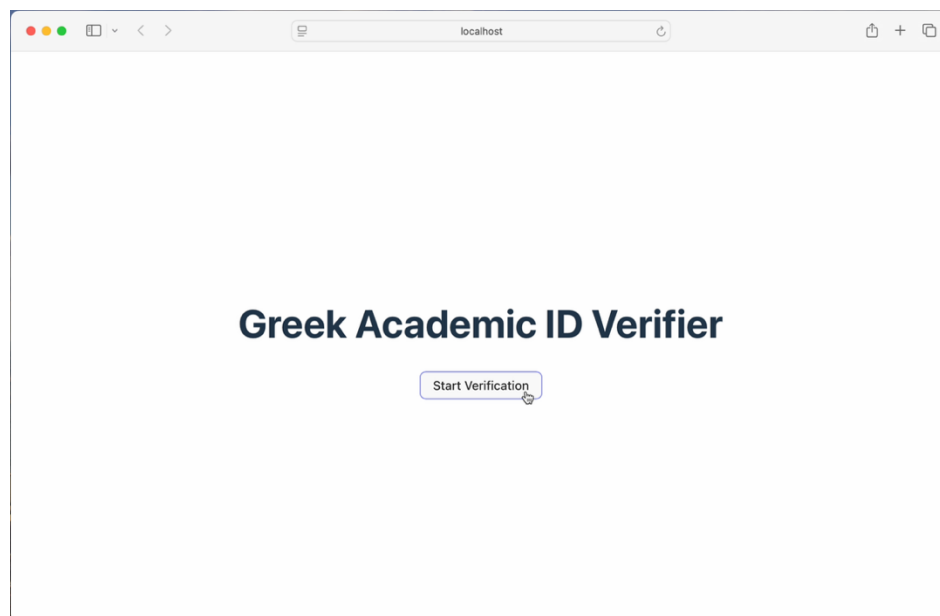
- Εκτέλεση της εφαρμογής:

“npm run dev”

(η εφαρμογή θα είναι διαθέσιμη τοπικά στην διεύθυνση: <http://localhost:5173/>)

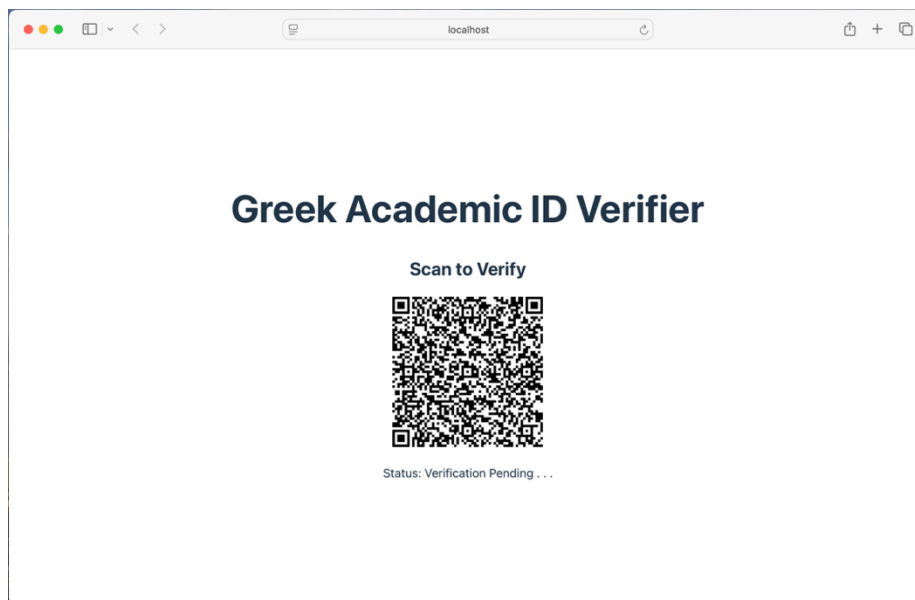
4.3.4 Παράδειγμα Εκτέλεσης

Κατά την εκκίνηση της εφαρμογής στη διεύθυνση του localhost, εμφανίζεται η σελίδα έναρξης.



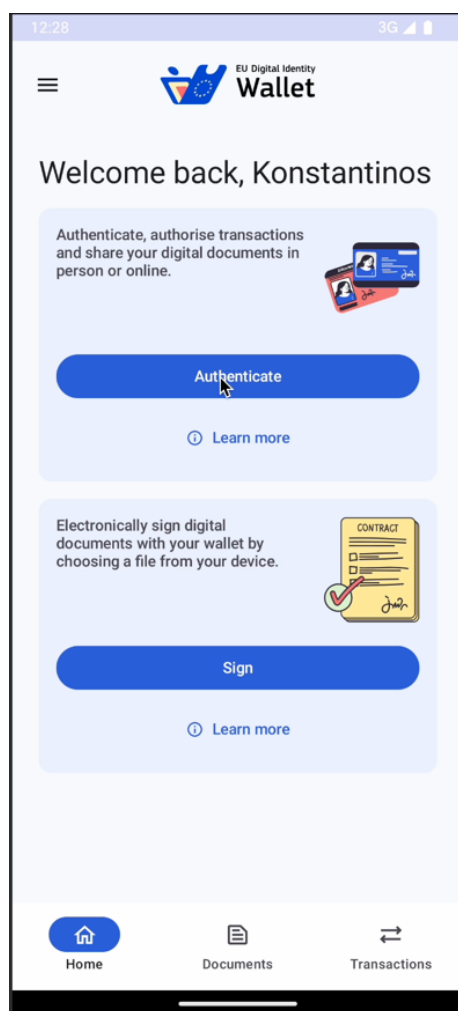
Εικόνα 4.3.4 – Αρχική σελίδα του Verifier.

Πατώντας το κουμπί: “Start Verification”, εμφανίζεται το QR-Code, το οποίο πρέπει να σκαναριστεί από την εφαρμογή του Wallet για να πραγματοποιηθεί η διαδικασία της επαλήθευσης.

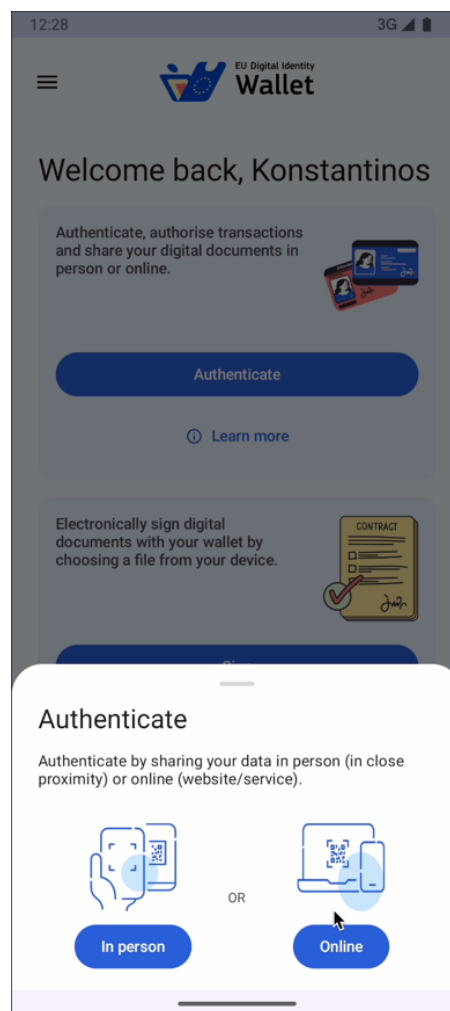


Εικόνα 4.3.5 – Σελίδα εμφάνισης QR-Code.

Στην συνέχεια γίνεται εκκίνηση της εφαρμογής του Wallet όπως παρουσιάζεται στην ενότητα 4.2.4 και στην αρχική σελίδα επιλέγεται το: “Authenticate”. Με την επιλογή αυτή εμφανίζονται οι δύο τρόποι επαλήθευσης που υποστηρίζονται από την εφαρμογή.



Εικόνα 4.3.6 – Αρχική οθόνη του Wallet.

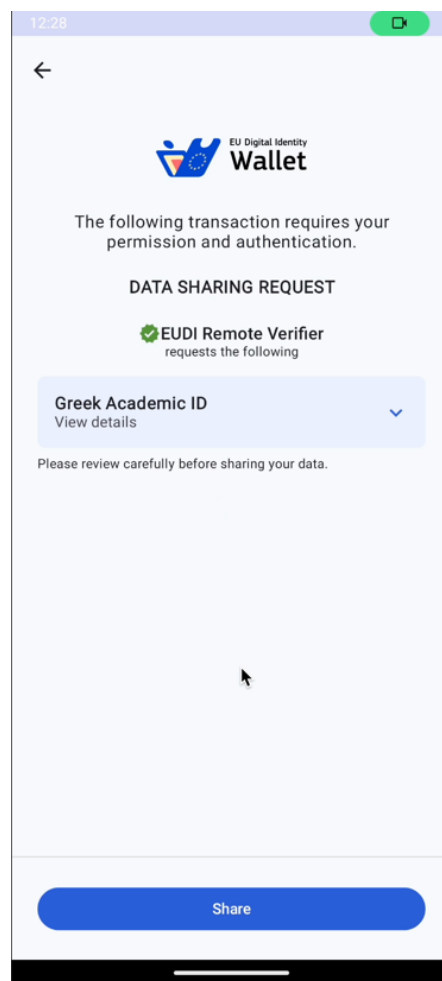


Εικόνα 4.3.7 – Οθόνη επιλογής τρόπου αυθεντικοποίησης.

Στην περίπτωση του συγκεκριμένου επαληθευτή γίνεται η επιλογή της “Online” επαλήθευσης. Με την επιλογή αυτή ανοίγει η ενσωματωμένη κάμερα της εφαρμογής για σκανάρισμα του QR-Code. Αμέσως μετά το σκανάρισμα εμφανίζεται το αίτημα κοινοποίησης των δεδομένων.

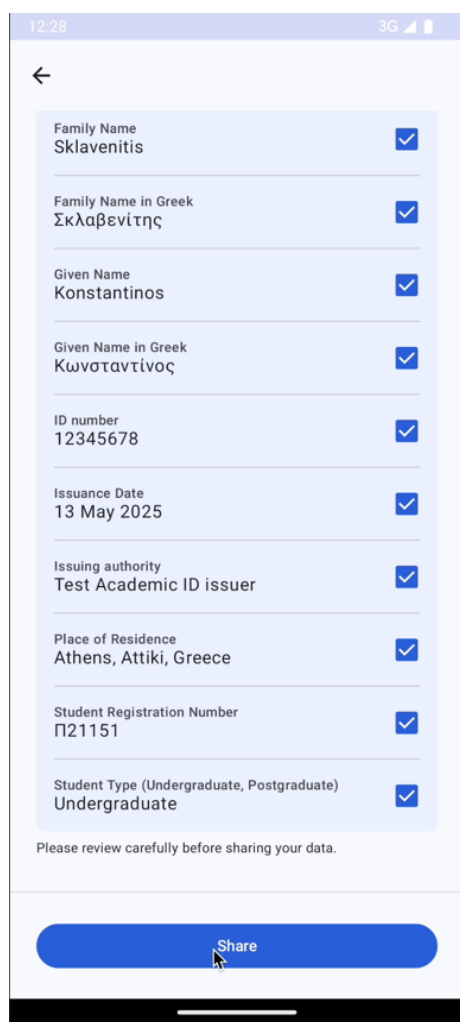


Εικόνα 4.3.8 – Οθόνη ενσωματωμένης κάμερας για σκανάρισμα.



Εικόνα 4.3.9 – Οθόνη προβολής αιτήματος επαλήθευσης.

Το αίτημα συνοδεύεται από τα στοιχεία τα οποία ζητάει ο επαληθευτής με σκοπό να τα δει ο χρήστης και να επιλέξει ποιά από αυτά θέλει να κοινοποιήσει. Στη συνέχεια, πατώντας το κουμπί: “Share”, ζητείται ο κωδικός PIN του χρήστη με σκοπό να εγκριθεί η συναλλαγή των στοιχείων για επαλήθευση.



12:28 3G

←

Family Name
Sklavenitis ✓

Family Name in Greek
Σκλαβενίτης ✓

Given Name
Konstantinos ✓

Given Name in Greek
Κωνσταντίνος ✓

ID number
12345678 ✓

Issuance Date
13 May 2025 ✓

Issuing authority
Test Academic ID issuer ✓

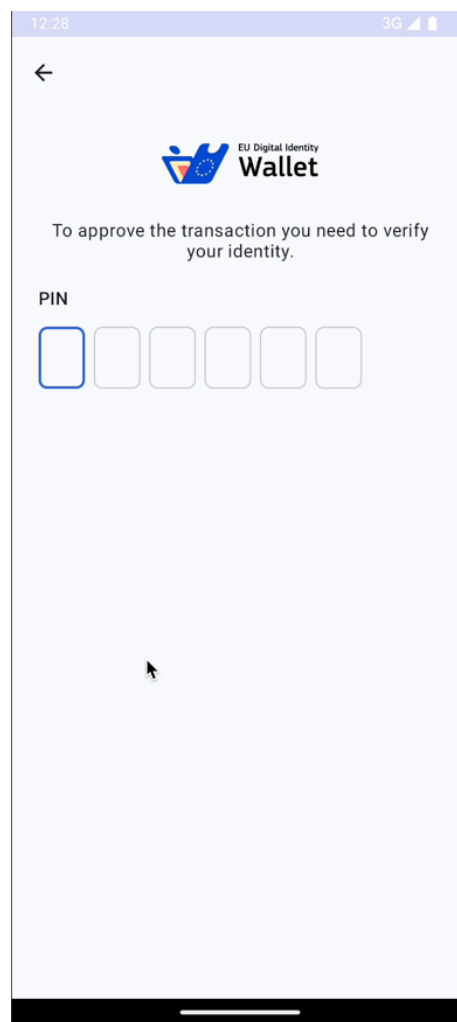
Place of Residence
Athens, Attiki, Greece ✓

Student Registration Number
Π21151 ✓

Student Type (Undergraduate, Postgraduate)
Undergraduate ✓

Please review carefully before sharing your data.

Share



12:28 3G

←

EU Digital Identity
Wallet

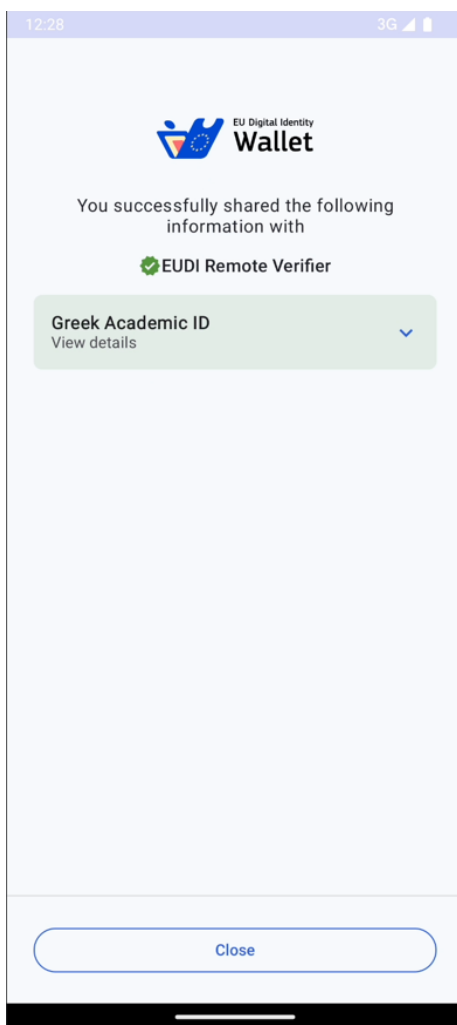
To approve the transaction you need to verify
your identity.

PIN

□ □ □ □ □ □

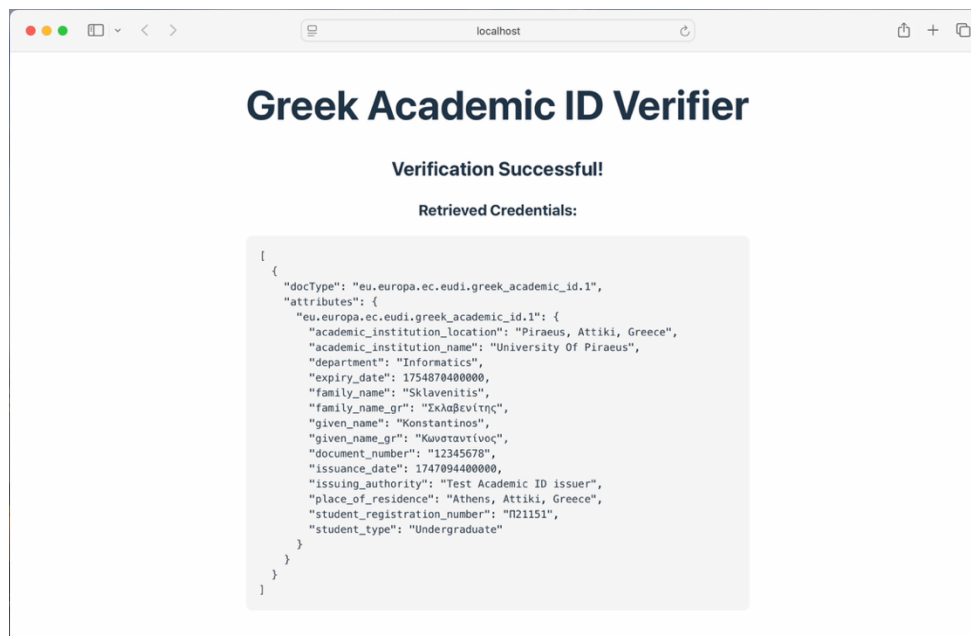
Εικόνα 4.3.10 – Οθόνη επιλογής στοιχείων για κοινοποίηση. **Εικόνα 4.3.11** – Οθόνη εισαγωγής κωδικού PIN για έγκριση.

Μετά την υποβολή του κωδικού εμφανίζεται μήνυμα επιτυχούς κοινοποίησης των στοιχείων στον επαληθευτή.



Εικόνα 4.3.12 – Οθόνη ενημέρωσης επιτυχούς κοινοποίησης στοιχείων.

Τέλος, επιστρέφοντας πίσω στην σελίδα του επαληθευτή, εμφανίζεται μήνυμα επιτυχούς επαλήθευσης, συνοδευόμενο από τα στοιχεία του διαπιστευτηρίου που ανακτήθηκαν.



Εικόνα 4.3.13 – Σελίδα ανάκτησης στοιχείων από το Wallet.

Κεφάλαιο 5^ο

5 Συμπεράσματα

Η παρούσα εργασία ανέλυσε το θέμα των ψηφιακών ταυτοτήτων μέσω του προτύπου Self-Sovereign Identity (SSI) και των Verifiable Credentials (VCs), μέσω της θεωρητικής εμβάθυνσης και πρακτικής υλοποίησης στο οικοσύστημα του European Digital Identity (EUDI). Παρουσιάστηκαν τα βασικά πρότυπα όπως τα SSI, VCs, VPs, OpenID4VC, Presentation Definitions και DCQL, ενώ αναλύθηκαν ζητήματα συμμόρφωσης με τους κανονισμούς eIDAS 2.0 και GDPR και η σημασία της αποκέντρωσης του ελέγχου από τον χρήστη.

Στο πρακτικό σκέλος, επιτεύχθηκε η έκδοση, υποστήριξη και επαλήθευση της Ελληνικής Ακαδημαϊκής Ταυτότητας ως διαπιστευτήριο στο EUDI Ecosystem, μέσω της επέκτασης του Issuer, της παραμετροποίησης του Wallet και της δημιουργίας ενός custom Verifier. Η υλοποίηση κάλυψε πλήρως τις απαιτήσεις του κύκλου ζωής ενός διαπιστευτηρίου (credential lifecycle), από την έκδοση έως την επαλήθευση, με βάση το πρότυπο OpenID4VC και τις σχετικές τεχνικές προδιαγραφές.

Από την αξιολόγηση της εργασίας προέκυψαν τα εξής συμπεράσματα:

- Το μοντέλο της ψηφιακής ταυτότητας και τα πρότυπα που το συνοδεύουν, αποτελούν λειτουργικές και πολλά υποσχόμενες τεχνολογίες, ωστόσο βρίσκονται ακόμη στο πρώιμο στάδιο της ανάπτυξης και απαιτούνται πολλές ενέργειες για την υιοθέτηση τους από τα κράτη-μέλη της Ε.Ε.
- Η επεκτασιμότητα της αρχιτεκτονικής του EUDI επιτρέπει την υλοποίηση custom σεναρίων, όπως η ένταξη εθνικών διαπιστευτηρίων, υπό την προϋπόθεση ύπαρξης εναρμονισμένων trust frameworks και metadata services.
- Τα πρότυπα του Selective Disclosure (επιλεκτική αποκάλυψης) και η προσέγγιση Privacy by Design, ενισχύουν την συμμόρφωση με τον κανονισμό GDPR και το νέο eIDAS (2.0), ελαχιστοποιώντας έτσι την αποκάλυψη προσωπικών δεδομένων και ενισχύοντας την προστασία των Ευρωπαίων πολιτών στον κυβερνοχώρο.
- Παρότι τα λογισμικά ανοιχτού κώδικα της Ε.Ε. παρέχουν λειτουργική βάση, απαιτείται καλή κατανόηση των προτύπων, των πρωτοκόλλων και των τεχνολογιών για να μπορέσει κάποιος να τα προσαρμόσει επιτυχώς σε custom περιβάλλοντα και νέους τύπους διαπιστευτηρίων.

Η συνολική εμπειρία έδειξε ότι η τεχνολογία SSI είναι εφαρμόσιμη σε πραγματικά σενάρια, υπό την προϋπόθεση συνεχούς συντονισμού σε Ευρωπαϊκό και Εθνικό επίπεδο. Η ενσωμάτωση της Ελληνικής Ακαδημαϊκής Ταυτότητας στο EUDI αποδεικνύει την ευελιξία του συστήματος και την εφικτότητα της μετάβασης σε ένα καθολικό, αποκεντρωμένο και διαλειτουργικό μοντέλο ψηφιακής ταυτοποίησης.

Κεφάλαιο 6^ο

6 Βιβλιογραφικές Πηγές

1. **Anna Zafeiropoulou, Evangelos Sakkopoulos**, “*Harmonising Digital Identity Documents. IISA 2023: 1-8*”, 2023
2. **Eleni Papadopoulou, Evangelos Sakkopoulos**, “*Semantic Cataloging of Public Services Using Basic Government Vocabularies and the Data Catalog Vocabulary for a Unified European Digital Market. IISA 2023: 1-4*”, 2023
3. **Panteleimon Krasadakis, Evangelos Sinos, Vassilios S. Verykios, Evangelos Sakkopoulos**, “*Efficient Named Entity Recognition on Greek Legislation. IISA 2022: 1-8*”, 2022
4. **Wikipedia**, “*Self-sovereign identity*”, 2025,
https://en.wikipedia.org/wiki/Self-sovereign_identity
5. **Sovrin Foundation**, “*Principles Of SSI V3*”, 2022,
<https://sovrin.org/principles-of-ssi/>
6. **European Union**, “*General Data Protection Regulation (EU 2016/679 – GDPR)*”, 2016,
<https://eur-lex.europa.eu/eli/reg/2016/679/oj/eng>
7. **European Commission**, “*Electronic Identification, Authentication and Trust Services Regulation (EU 2024/1183 – eIDAS 2.0)*”, 2024,
<https://eur-lex.europa.eu/eli/reg/2024/1183/oj/eng>
8. **European Commission**, “*EIDAS SUPPORTED SELF-SOVEREIGN IDENTITY*”, 2019,
https://ec.europa.eu/futurium/en/system/files/ged/eidas_supported_ssi_may_2019_0.pdf
9. **W3C**, “*Verifiable Credentials Data Model 2.0*”, 2023,
<https://www.w3.org/TR/vc-data-model-2.0/>
10. **W3C**, “*Securing Verifiable Credentials using JOSE and COSE*”, 2023,
<https://www.w3.org/TR/vc-jose-cose/>
11. **Wikipedia**, “*Verifiable credentials*”, 2025,
https://en.wikipedia.org/wiki/Verifiable_credentials
12. **OpenID Foundation**, “*OpenID for Verifiable Credentials - Overview*”, 2025,
<https://openid.net/sg/openid4vc/>
13. **OpenID Foundation**, “*OpenID for Verifiable Credential Issuance - draft 16*”, 2025,
https://openid.net/specs/openid-4-verifiable-credential-issuance-1_0.html
14. **OpenID Foundation**, “*OpenID for Verifiable Presentations 1.0 Final*”, 2025,
https://openid.net/specs/openid-4-verifiable-presentations-1_0-final.html
15. **W3C CCG**, “*Verifiable Credentials API v0.7*”, 2025,
<https://w3c-ccg.github.io/vc-api/>
16. **OpenID Foundation**, “*OpenID4VC High Assurance Interoperability Profile - draft 03*”, 2025,

- https://openid.net/specs/openid4vc-high-assurance-interoperability-profile-1_0-03.html
17. **Decentralized Identity Foundation (DIF)**, “*Presentation Exchange*”, 2023, <https://identity.foundation/presentation-exchange/>
 18. **Walt.id**, “*How to implement presentation_definition in wallets*”, 2025, <https://docs.walt.id/community-stack/wallet/api/credentials/guides/present-w3c-mdl-oid4vp-external-signatures>
 19. **OpenID Foundation**, “*OpenID for Verifiable Presentations 1.0 – 6. Digital Credentials Query Language (DCQL)*”, 2025, https://openid.net/specs/openid-4-verifiable-presentations-1_0.html#name-digital-credentials-query-l
 20. **European Commission**, “*EUDI – GitHub Repository*”, 2025, <https://github.com/eu-digital-identity-wallet>
 21. **European Union**, “*ABOUT DC4EU*”, 2025, <https://www.dc4eu.eu/project/wp6/>
 22. **European Commission**, “*EUDIW Architecture and Reference Framework - ARF*”, 2025, <https://eu-digital-identity-wallet.github.io/eudi-doc-architecture-and-reference-framework/2.0.0/>
 23. **Zahra Ebadi Ansaroudi et al.**, “*Navigating secure storage requirements for EUDI Wallets: a review paper*”, SpringerOpen, 2025, <https://jis-urasipjournals.springeropen.com/articles/10.1186/s13635-025-00187-6>
 24. **Visual Studio Code**, “*The Open Source AI Coding Editor*”, <https://code.visualstudio.com>
 25. **Python**, “*Python 3.10 documentation*”, <https://docs.python.org/3.10/>
 26. **Flask**, “*Flask Documentation*”, <https://flask.palletsprojects.com/en/stable/>
 27. **Ngrok**, “*API’s Online*”, <https://ngrok.com>
 28. **Android Studio**, “*The official IDE for Android app development*”, <https://developer.android.com/studio>
 29. **Kotlin**, “*Kotlin docs*”, <https://kotlinlang.org/docs/home.html>
 30. **SwaggerUI**, “*EUDIW Verifier Endpoints*”, <https://dev.verifier-backend.eudiw.dev/swagger-ui>
 31. **NodeJS**, “*Run JavaScript Everywhere*”, <https://nodejs.org/en>
 32. **NPM**, “*Build amazing things*”, <https://www.npmjs.com>
 33. **Vite**, “*The Build Tool for the Web*”, <https://vite.dev>
 34. **JavaScript**, “*Programming Language*”, <https://www.javascript.com>
 35. **React**, “*The library for web and native user interfaces*”, <https://react.dev>

36. **StackOverflow**, “*Q&A Platform for resolving programming issues*”,
<https://stackoverflow.com/questions>