ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΙΡΑΙΩΣ – ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ

Πρόγραμμα Μεταπτυχιακών Σπουδών

«Προηγμένα Συστήματα Πληροφορικής – Ανάπτυξη Λογισμικού και Τεχνητής Νοημοσύνης»

Μεταπτυχιακή Διατριβή

Τίτλος Διατριβής	**(Ελληνικά)** Σύστημα έξυπνης διαμοίρασης αρχείων με post quantum ψηφιακές υπογραφές **(Αγγλικά)** Smart File Sharing System with Post-Quantum Digital Signatures
Ονοματεπώνυμο Φοιτητή	**Γεώργιος Σταμάτης**
Πατρώνυμο	**Παναγιώτης**
Αριθμός Μητρώου	**ΜΠΣΠ/24049**
Επιβλέπων	**Ευάγγελος Σακκόπουλος, Αναπληρωτής Καθηγητής**

Ημερομηνία Παράδοσης **Μάρτιος 2026**

Τριμελής Εξεταστική Επιτροπή

Ευάγγελος Σακκόπουλος	**Ευθύμιος Αλέπης**	**Ιωάννης Τασούλας**
Αναπληρωτής Καθηγητής	**Καθηγητής**	**Επικ. Καθηγητης**

Περίληψη

Η παρούσα μεταπτυχιακή διατριβή παρουσιάζει τον σχεδιασμό και την υλοποίηση ενός διαδικτυακού συστήματος για τη δημιουργία και επαλήθευση ψηφιακών υπογραφών σε έγγραφα PDF.

Το σύστημα υποστηρίζει τρεις τρόπους υπογραφής: κλασική RSA, post-quantum (Dilithium3) και υβριδική (RSA + Dilithium3), προκειμένου να εξασφαλίσει τόσο συμβατότητα με υπάρχοντα εργαλεία όσο και προστασία απέναντι σε μελλοντικές κβαντικές επιθέσεις. Κάθε χρήστης διαθέτει το δικό του ζεύγος κλειδιών (δημόσιο/ιδιωτικό) για RSA και Dilithium3, τα οποία δημιουργούνται και διαχειρίζονται μέσω ειδικής σελίδας της εφαρμογής.

Η υλοποίηση βασίζεται σε Jakarta EE (JSF, PrimeFaces), Apache PDFBox και Bouncy Castle. Η επαλήθευση των υπογραφών γίνεται με βάση το ByteRange του PDF, σύμφωνα με το πρότυπο PAdES. Το σύστημα προσφέρει πλήρες ιστορικό ενεργειών (audit), στατιστικά στοιχεία και δυνατότητα αποστολής υπογεγραμμένων εγγράφων μέσω email. Τα αποτελέσματα δείχνουν ότι το σύστημα λειτουργεί επιτυχώς και είναι συμβατό με τυπικούς PDF readers.

Abstract

This thesis presents the development of a web system for managing and verifying digital signatures in PDF documents. The system supports three signing modes: classical RSA, post-quantum (Dilithium3), and hybrid (RSA + Dilithium), to ensure both compatibility with existing tools and protection against future quantum attacks.Each user has their own key pair (public/private) for RSA and Dilithium3, which are created and managed through a dedicated application page.

The implementation is based on Java EE (Jakarta Faces, PrimeFaces), Apache PDFBox, and Bouncy Castle. Signature verification is performed according to the PDF ByteRange, in compliance with the PAdES standard. The system provides a full audit trail, statistics, and the ability to send signed documents via email. The main objectives were achieved and the system is functional and compatible with standard PDF readers.

Πίνακας Περιεχομένων

Περίληψη 3
Abstract 3
1. Εισαγωγή 6
1.1 Σκοπός και αντικείμενο της εργασίας 6
1.2 Στόχοι 6
1.3 Δομή της εργασίας 6
1.4 Συνεισφορά της εργασίας 6
2. Θεωρητικό Υπόβαθρο 8
2.1 Ψηφιακές υπογραφές 8
2.2 Ασύμμετρη Κρυπτογραφία 8
2.3 RSA και κλασική κρυπτογραφία 9
2.4 Post-quantum κρυπτογραφία και Dilithium 9
2.5 Υβριδικές υπογραφές 10
2.6 Ψηφιακές υπογραφές σε PDF (PAdES) 10
3. Ανάλυση Απαιτήσεων 11
3.1 Λειτουργικές απαιτήσεις 11
3.2 Μη λειτουργικές απαιτήσεις 11
4. Σχεδιασμός Συστήματος 13
4.1 Αρχιτεκτονική 13
4.2 Μοντέλο δεδομένων 13
4.3 Ροές λειτουργίας του συστήματος 14
4.3.1 Ροή Upload & Sign 15
4.3.2 Δημιουργία ψηφιακής υπογραφής 15
4.3.3 Ενσωμάτωση υπογραφής στο PDF 15
4.3.4 Υποστήριξη RSA, Dilithium και υβριδικών υπογραφών 16
4.3.5 Διαχείριση κρυπτογραφικών κλειδιών 18
4.4 Ροή Verify (Επαλήθευση υπογραφής) 18
4.4.1 Επιλογή εγγράφου 18
4.4.2 Επαλήθευση μέσω ByteRange 18
4.4.3 Επαλήθευση RSA και Dilithium 19
4.4.4 Fallback επαλήθευση μέσω hash 19
4.4.5 Αποτελέσματα επαλήθευσης 19
5. Υλοποίηση 21
5.1 Τεχνολογίες 21
5.2 Βασικές λειτουργίες 21
5.3 Δομή κώδικα 22
5.4 Εργαλεία Ανάπτυξης και Χρήση Εργαλείων Τεχνητής Νοημοσύνης 22
6. Αποτελέσματα και Δοκιμές 24
6.1 Περιβάλλον δοκιμών 24
6.2 Σενάρια δοκιμών 24
6.3 Αποτελέσματα 28
6.4 Συζήτηση αποτελεσμάτων 28
7. Συμπεράσματα 29
7.1 Αποτελέσματα 29
7.2 Μελλοντική Εργασία 29
8. Βιβλιογραφία 29

Λίστα Σχημάτων

Εικόνα 1 . Διαδικασία δημιουργίας και επαλήθευσης ψηφιακής υπογραφής με χρήση ασύμμετρης κρυπτογραφίας. 9
Εικόνα 2 . Αρχιτεκτονική του συστήματος δημιουργίας και επαλήθευσης ψηφιακών υπογραφών... 13
Εικόνα 3 . Διαδικασία δημιουργίας υβριδικής ψηφιακής υπογραφής PDF με χρήση RSA και Dilithium. 17
Εικόνα 4 . Οθόνη εισόδου χρήστη στο Σύστημα έξυπνης διαμοίρασης αρχείων με PQC ψηφιακές υπογραφές. 24
Εικόνα 5 . Διαδικασία υπογραφής εγγράφου PDF μέσω της εφαρμογής (RSA, Dilithium3 και υβριδικό σχήμα). 25
Εικόνα 6 . Οθόνη επαλήθευσης – επιλογή εγγράφου και εμφάνιση αποτελεσμάτων (RSA OK, PQC OK, overall). 25
Εικόνα 7 . Οθόνη αποτυχημένης επαλήθευσης 26
Εικόνα 8 . Οθόνη ιστορικού ενεργειών – λίστα εγγράφων και καταγραφή ενεργειών ανά έγγραφο. 26
Εικόνα 9 . Οθόνη στατιστικών – σύνολο εγγράφων, επαληθεύσεις και κατανομή τύπων υπογραφής. 27
Εικόνα 10 . Οθόνη δημιουργίας και ξεκλειδώματος προσωπικών κλειδιών (RSA + PQC) ανά χρήστη. 27

1. Εισαγωγή

1.1 Σκοπός και αντικείμενο της εργασίας

Τα τελευταία χρόνια, η ραγδαία αύξηση της ψηφιοποίησης εγγράφων έχει οδηγήσει στην ευρεία χρήση ηλεκτρονικών αρχείων σε πλήθος εφαρμογών, όπως η δημόσια διοίκηση, οι επιχειρήσεις και οι ηλεκτρονικές συναλλαγές. Στο πλαίσιο αυτό, η ανάγκη για διασφάλιση της αυθεντικότητας, της ακεραιότητας και της μη αποποίησης ευθύνης των εγγράφων καθίσταται ιδιαίτερα σημαντική.

Οι ψηφιακές υπογραφές αποτελούν βασικό μηχανισμό για την επίτευξη των παραπάνω στόχων, καθώς επιτρέπουν την επαλήθευση της προέλευσης ενός εγγράφου και την ανίχνευση οποιασδήποτε τροποποίησης μετά την υπογραφή του. Ωστόσο, η πλειονότητα των σύγχρονων συστημάτων βασίζονται σε κλασικούς αλγορίθμους κρυπτογραφίας, όπως το RSA, οι οποίοι θεωρούνται ασφαλείς με βάση τα σημερινά δεδομένα.

Η εμφάνιση όμως των κβαντικών υπολογιστών αναμένεται να επηρεάσει σημαντικά την ασφάλεια των υπαρχόντων κρυπτογραφικών συστημάτων, καθώς αλγόριθμοι όπως ο Shor δύνανται θεωρητικά να παραβιάσουν την ασφάλεια του RSA. Αυτό δημιουργεί την ανάγκη μετάβασης σε νέες μορφές κρυπτογραφίας, γνωστές ως post-quantum cryptography (PQC), οι οποίες είναι σχεδιασμένες ώστε να παραμένουν ασφαλείς ακόμη και απέναντι σε κβαντικές επιθέσεις.

Συνεπώς, προκύπτει ένα σημαντικό πρόβλημα: η μετάβαση αυτή δεν μπορεί να γίνει άμεσα, καθώς τα υπάρχοντα συστήματα και εργαλεία βασίζονται σε κλασικούς αλγορίθμους και απαιτούν συμβατότητα προς τα πίσω.

Η παρούσα εργασία έρχεται να καλύψει αυτό το κενό, προτείνοντας και υλοποιώντας ένα σύστημα διαχείρισης και επαλήθευσης ψηφιακών υπογραφών σε έγγραφα PDF, το οποίο υποστηρίζει τόσο κλασικές υπογραφές (RSA) όσο και μετα-κβαντικές υπογραφές (Dilithium3), καθώς και υβριδικά σχήματα που συνδυάζουν και τα δύο.

Με τον τρόπο αυτό επιτυγχάνεται αφενός η συμβατότητα με τα υπάρχοντα συστήματα και αφετέρου η προετοιμασία για την εποχή της μετα-κβαντικής κρυπτογραφίας.

1.2 Στόχοι

Οι βασικοί στόχοι της παρούσας εργασίας συνοψίζονται ως εξής:

- Σχεδιασμός web εφαρμογής για ανέβασμα, υπογραφή και επαλήθευση PDF.
- Υποστήριξη τριών τρόπων υπογραφής: RSA, Dilithium (PQC), υβριδική (RSA + PQC).
- Επαλήθευση υπογραφών με βάση το ByteRange του PDF (σύμφωνα με το PDF standard).
- Καταγραφή ιστορικού ενεργειών (audit) και παρουσίαση στατιστικών.
- Αποστολή υπογεγραμμένων εγγράφων μέσω email.
- Διαχείριση προσωπικών κλειδιών ανά χρήστη: δημιουργία και χρήση ζεύγους κλειδιών

(RSA + Dilithium3) ανά χρήστη, με ξεκλείδωμα keystore ανά συνεδρία και δυνατότητα επανέκδοσης κλειδιών.

1.3 Δομή της εργασίας

Η εργασία οργανώνεται σε οκτώ κεφάλαια. Μετά την εισαγωγή, το Κεφάλαιο 2 παρουσιάζει το θεωρητικό υπόβαθρο (ψηφιακές υπογραφές, RSA, PQC, Dilithium). Το Κεφάλαιο 3 περιγράφει την ανάλυση απαιτήσεων, το Κεφάλαιο 4 τον σχεδιασμό του συστήματος και το Κεφάλαιο 5 την υλοποίηση. Το Κεφάλαιο 6 αφορά τα αποτελέσματα και τις δοκιμές, το Κεφάλαιο 7 τα συμπεράσματα και το Κεφάλαιο 8 τη βιβλιογραφία.

1.4 Συνεισφορά της εργασίας

Η συμβολή της παρούσας εργασίας εντοπίζεται τόσο σε ερευνητικό όσο και σε πρακτικό επίπεδο. Συγκεκριμένα, η εργασία συμβάλλει στην ανάπτυξη ενός συστήματος ψηφιακών υπογραφών που συνδυάζει κλασικές και μετα-κβαντικές τεχνικές κρυπτογραφίας.

Συγκεκριμένα, οι βασικές συνεισφορές της εργασίας είναι:

- Η υλοποίηση ενός web συστήματος για δημιουργία και επαλήθευση ψηφιακών υπογραφών σε έγγραφα PDF.
- Η ενσωμάτωση του μετα-κβαντικού αλγορίθμου Dilithium3 σε ένα περιβάλλον υπογραφής εγγράφων.
- Η υποστήριξη υβριδικών υπογραφών RSA και Dilithium πάνω στα ίδια δεδομένα ByteRange.
- Η διατήρηση συμβατότητας με υπάρχοντα PDF readers μέσω CMS υπογραφών.
- Η υλοποίηση μηχανισμού επαλήθευσης που υποστηρίζει τόσο κλασικές όσο και PQC υπογραφές.

2. Θεωρητικό Υπόβαθρο

2.1 Ψηφιακές υπογραφές

Στην ψηφιακή εποχή, η ανάγκη για ασφαλή ανταλλαγή εγγράφων είναι ιδιαίτερα αυξημένη. Σε αντίθεση με τα έντυπα έγγραφα, όπου η υπογραφή ενός ατόμου μπορεί να επαληθευτεί οπτικά, τα ηλεκτρονικά έγγραφα απαιτούν ειδικούς μηχανισμούς για την εξασφάλιση της αυθεντικότητας και της ακεραιότητάς τους. Για παράδειγμα, όταν ένας χρήστης υπογράφει ψηφιακά ένα αρχείο PDF, δημιουργείται μια μοναδική ψηφιακή τιμή (signature) που εξαρτάται τόσο από το περιεχόμενο του εγγράφου όσο και από το ιδιωτικό κλειδί του χρήστη. Εάν το έγγραφο τροποποιηθεί έστω και ελάχιστα, η υπογραφή δεν θα είναι πλέον έγκυρη. Η ψηφιακή υπογραφή αποτελεί την κρυπτογραφική ισοδύναμη της χειρόγραφης υπογραφής, επιτρέποντας την ασφαλή επιβεβαίωση της ταυτότητας του υπογράφοντος και την ακεραιότητα των δεδομένων.

Οι ψηφιακές υπογραφές εξασφαλίζουν τα εξής:

- **Αυθεντικότητα:** Ο παραλήπτης μπορεί να επαληθεύσει ότι το έγγραφο προέρχεται από τον υπογράφοντα.
- **Ακεραιότητα:** Κάθε τροποποίηση του εγγράφου μετά την υπογραφή καθιστά την επαλήθευση ανεπιτυχή.
- **Μη αποποίηση ευθύνης (Non-repudiation):** Ο υπογράφων δεν μπορεί να αρνηθεί ότι υπέγραψε το έγγραφο.

Οι ψηφιακές υπογραφές βασίζονται σε ασύμμετρη κρυπτογραφία: ο υπογράφων χρησιμοποιεί ιδιωτικό κλειδί για να δημιουργήσει την υπογραφή, ενώ οποιοσδήποτε μπορεί να την επαληθεύσει με το δημόσιο κλειδί. Στην επόμενη ενότητα παρουσιάζεται η αρχή λειτουργίας της ασύμμετρης κρυπτογραφίας και η σχέση μεταξύ δημόσιου και ιδιωτικού κλειδιού.

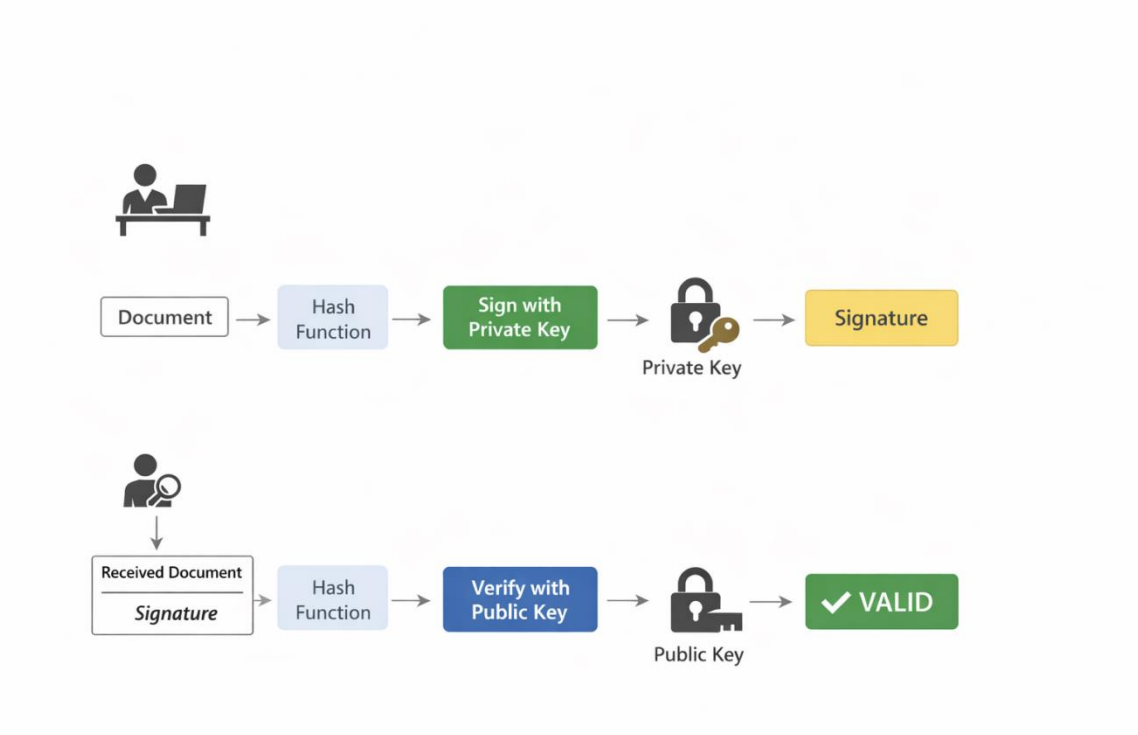
2.2 Ασύμμετρη Κρυπτογραφία

Η ασύμμετρη κρυπτογραφία αποτελεί τη βάση των ψηφιακών υπογραφών και χρησιμοποιείται ευρέως σε εφαρμογές ασφάλειας πληροφοριών, όπως το HTTPS, τα ψηφιακά πιστοποιητικά και τα συστήματα αυθεντικοποίησης.

Η βασική λειτουργία της ασύμμετρης κρυπτογραφίας μπορεί να περιγραφεί με τα εξής βήματα:

- Δημιουργία ζεύγους κλειδιών (δημόσιο και ιδιωτικό).
- Υπογραφή δεδομένων με χρήση του ιδιωτικού κλειδιού.
- Επαλήθευση της υπογραφής με χρήση του δημόσιου κλειδιού.

Το δημόσιο κλειδί μπορεί να διανεμηθεί ελεύθερα, ενώ το ιδιωτικό κλειδί παραμένει μυστικό στον κάτοχό του. Η ασφάλεια των συστημάτων αυτών βασίζεται σε μαθηματικές συναρτήσεις μονής κατεύθυνσης (one-way functions), οι οποίες είναι εύκολο να υπολογιστούν αλλά πρακτικά αδύνατο να αντιστραφούν.



Εικόνα 1. Διαδικασία δημιουργίας και επαλήθευσης ψηφιακής υπογραφής με χρήση ασύμμετρης κρυπτογραφίας.

2.3 RSA και κλασική κρυπτογραφία

Το RSA (Rivest–Shamir–Adleman) είναι ένα από τα πιο διαδεδομένα αλγόριθμα ασύμμετρης κρυπτογραφίας. Βασίζεται στη δυσκολία παραγοντοποίησης μεγάλων ακεραίων. Για υπογραφές, συνήθως χρησιμοποιείται το σχήμα RSA-PSS ή RSA με PKCS#1 v1.5 σε συνδυασμό με hash (π.χ. SHA-256).

Το RSA θεωρείται ασφαλές με σημερινά μέσα, αλλά οι κβαντικοί υπολογιστές με αλγόριθμο Shor θα μπορούν να παραγοντοποιήσουν αποτελεσματικά και να παραβιάσουν την ασφάλεια του RSA. Ο αλγόριθμος RSA χρησιμοποιείται ευρέως σε πλήθος εφαρμογών ασφάλειας, όπως τα ψηφιακά πιστοποιητικά (X.509), τα πρωτόκολλα ασφαλούς επικοινωνίας (TLS/SSL) και οι ψηφιακές υπογραφές εγγράφων.

2.4 Post-quantum κρυπτογραφία και Dilithium

Η εμφάνιση των κβαντικών υπολογιστών αποτελεί μία από τις σημαντικότερες προκλήσεις για την κρυπτογραφία. Σε αντίθεση με τους κλασικούς υπολογιστές, οι κβαντικοί υπολογιστές μπορούν να επιλύουν συγκεκριμένα μαθηματικά προβλήματα πολύ πιο αποδοτικά, καθιστώντας ευάλωτους πολλούς από τους υπάρχοντες αλγορίθμους. Για τον λόγο αυτό, η ερευνητική κοινότητα έχει στραφεί στην ανάπτυξη αλγορίθμων post-quantum cryptography, οι οποίοι δεν βασίζονται σε προβλήματα που μπορούν να λυθούν αποδοτικά από κβαντικούς υπολογιστές.

Συνεπώς, η post-quantum κρυπτογραφία (PQC) αναφέρεται σε αλγορύθμους που θεωρούνται ασφαλείς ακόμη και απέναντι σε κβαντικούς υπολογιστές. Το NIST διεξήγαγε διαδικασία τυποποίησης και το 2022 ανακήρυξε νικητές για υπογραφές, μεταξύ των οποίων το **Dilithium**.

Το Dilithium βασίζεται σε δομές πλεγμάτων (lattice-based cryptography) και είναι ένα από τα προτεινόμενα σχήματα υπογραφής στο FIPS 204. Προσφέρει ισοδύναμη ασφάλεια με το RSA αλλά με

ανθεκτικότητα σε κβαντικές επιθέσεις.

2.5 Υβριδικές υπογραφές

Για λόγους συμβατότητας και σταδιακής μετάβασης, πολλά συστήματα χρησιμοποιούν **υβριδικές** υπογραφές: υπογράφουν το ίδιο περιεχόμενο και με RSA και με PQC. Έτσι, το έγγραφο παραμένει επαληθεύσιμο τόσο με κλασικά όσο και με PQC εργαλεία. Η χρήση υβριδικών υπογραφών αποτελεί πρακτική προσέγγιση για τη μετάβαση από τα κλασικά στα post-quantum συστήματα. Με τον τρόπο αυτό, ένα σύστημα μπορεί να συνεχίσει να είναι συμβατό με τα υπάρχοντα εργαλεία, ενώ παράλληλα ενσωματώνει μηχανισμούς ασφάλειας για το μέλλον.

2.6 Ψηφιακές υπογραφές σε PDF (PAdES)

Οι ψηφιακές υπογραφές σε αρχεία PDF βασίζονται σε συγκεκριμένες δομές του αρχείου, οι οποίες επιτρέπουν την ενσωμάτωση της υπογραφής χωρίς να αλλοιώνεται το περιεχόμενο. Το PAdES (PDF Advanced Electronic Signatures) είναι το ETSI standard που ορίζει πώς οι υπογραφές εφαρμόζονται σε PDF. Ένα από τα βασικά στοιχεία αυτής της διαδικασίας είναι το ByteRange.

Το ByteRange ορίζει ποια ακριβώς τμήματα του PDF συμμετέχουν στη διαδικασία υπογραφής. Με τον τρόπο αυτό εξασφαλίζεται ότι η υπογραφή καλύπτει το περιεχόμενο του εγγράφου, εξαιρώντας μόνο το πεδίο στο οποίο αποθηκεύεται η ίδια η υπογραφή.

Με απλά λόγια, καθορίζει ποια bytes του αρχείου υπογράφονται (συνήθως το περιεχόμενο πριν και μετά το πεδίο /Contents της υπογραφής). Η επαλήθευση πρέπει να διαβάσει ακριβώς αυτά τα bytes και να ελέγξει την υπογραφή πάνω σε αυτά.

Στο κεφάλαιο αυτό παρουσιάστηκαν οι βασικές έννοιες που σχετίζονται με τις ψηφιακές υπογραφές και την κρυπτογραφία, οι οποίες αποτελούν το θεωρητικό υπόβαθρο για την κατανόηση της υλοποίησης του συστήματος που παρουσιάζεται στα επόμενα κεφάλαια.

3. Ανάλυση Απαιτήσεων

3.1 Λειτουργικές απαιτήσεις

Οι λειτουργικές απαιτήσεις περιγράφουν τις βασικές δυνατότητες που πρέπει να παρέχει το σύστημα στους χρήστες του. Οι απαιτήσεις αυτές καθορίζουν τις ενέργειες που μπορεί να εκτελέσει ένας χρήστης, καθώς και τις βασικές λειτουργίες που υποστηρίζει η εφαρμογή.

Πίνακας 1: Λειτουργικές απαιτήσεις συστήματος

Αριθμός	Περιγραφή	Προτεραιότητα
R1	Δυνατότητα ανέβασματος αρχείου PDF	Υψηλή
R2	Δυνατότητα υπογραφής PDF με RSA, Dilithium ή υβριδικά	Υψηλή
R3	Η υπογραφή να εμφανίζεται οπτικά στο έγγραφο	Μέτρια
R4	Ο χρήστης να μπορεί να επαληθεύσει την υπογραφή ενός εγγράφου	Υψηλή
R5	Η επαλήθευση να βασίζεται στο ByteRange του PDF	Υψηλή
R6	Ο χρήστης να μπορεί να στείλει το υπογεγραμμένο PDF μέσω email	Μέτρια
R7	Καταγραφή όλων των ενεργειών (upload, sign, verify, email, download)	Υψηλή
R8	Παρουσίαση στατιστικών (έγγραφα, events, κατανομές)	Μέτρια
R9	Έλεγχος πρόσβασης ανά σελίδα (login, δικαιώματα)	Υψηλή
R10	Δυνατότητα αυθεντικοποίησης χρήστη μέσω Google (OAuth 2.0)	Μέτρια

Οι παραπάνω λειτουργικές απαιτήσεις καλύπτουν τον βασικό κύκλο ζωής ενός εγγράφου στο σύστημα, ο οποίος περιλαμβάνει την αποστολή, την υπογραφή, την επαλήθευση και τη διαχείριση του εγγράφου. Ιδιαίτερη έμφαση δίνεται στη διαδικασία επαλήθευσης, η οποία βασίζεται στο ByteRange του PDF, εξασφαλίζοντας υψηλό επίπεδο ασφάλειας και αξιοπιστίας.

3.2 Μη λειτουργικές απαιτήσεις

Οι μη λειτουργικές απαιτήσεις αφορούν τα ποιοτικά χαρακτηριστικά του συστήματος και καθορίζουν περιορισμούς και ιδιότητες που σχετίζονται με την απόδοση, την ασφάλεια και τη συνολική ποιότητα και αξιοπιστία της εφαρμογής.

- **Ασφάλεια:** Η προστασία των κρυπτογραφικών κλειδιών αποτελεί κρίσιμο στοιχείο του συστήματος. Τα ιδιωτικά κλειδιά δεν αποθηκεύονται στη βάση δεδομένων αλλά σε ασφαλές περιβάλλον

στο filesystem, ενώ η πρόσβαση σε αυτά ελέγχεται μέσω μηχανισμών αυθεντικοποίησης. Επιπλέον, ευαίσθητα δεδομένα όπως credentials διαχειρίζονται με ασφαλή τρόπο.

- **Απόδοση:** Το σύστημα πρέπει να μπορεί να επεξεργάζεται την υπογραφή και την επαλήθευση εγγράφων σε εύλογο χρονικό διάστημα, ώστε να εξασφαλίζεται καλή εμπειρία χρήστη. Για τυπικά έγγραφα PDF, ο χρόνος εκτέλεσης δεν θα πρέπει να υπερβαίνει τα 10 δευτερόλεπτα.
- **Συμβατότητα:** Τα παραγόμενα υπογεγραμμένα έγγραφα πρέπει να είναι συμβατά με ευρέως χρησιμοποιούμενα εργαλεία ανάγνωσης PDF (όπως Adobe Acrobat Reader), ώστε να διασφαλίζεται η διαλειτουργικότητα του συστήματος.
- **Επεκτασιμότητα**: Το σύστημα θα πρέπει να είναι σχεδιασμένο με τρόπο που να επιτρέπει την εύκολη προσθήκη νέων αλγορίθμων υπογραφής ή λειτουργιών στο μέλλον, χωρίς σημαντικές αλλαγές στην αρχιτεκτονική του.

Οι παραπάνω μη λειτουργικές απαιτήσεις διασφαλίζουν ότι το σύστημα δεν είναι μόνο λειτουργικά πλήρες, αλλά και αξιόπιστο, αποδοτικό και επεκτάσιμο, καλύπτοντας τις σύγχρονες ανάγκες ασφάλειας και διαχείρισης ψηφιακών εγγράφων.

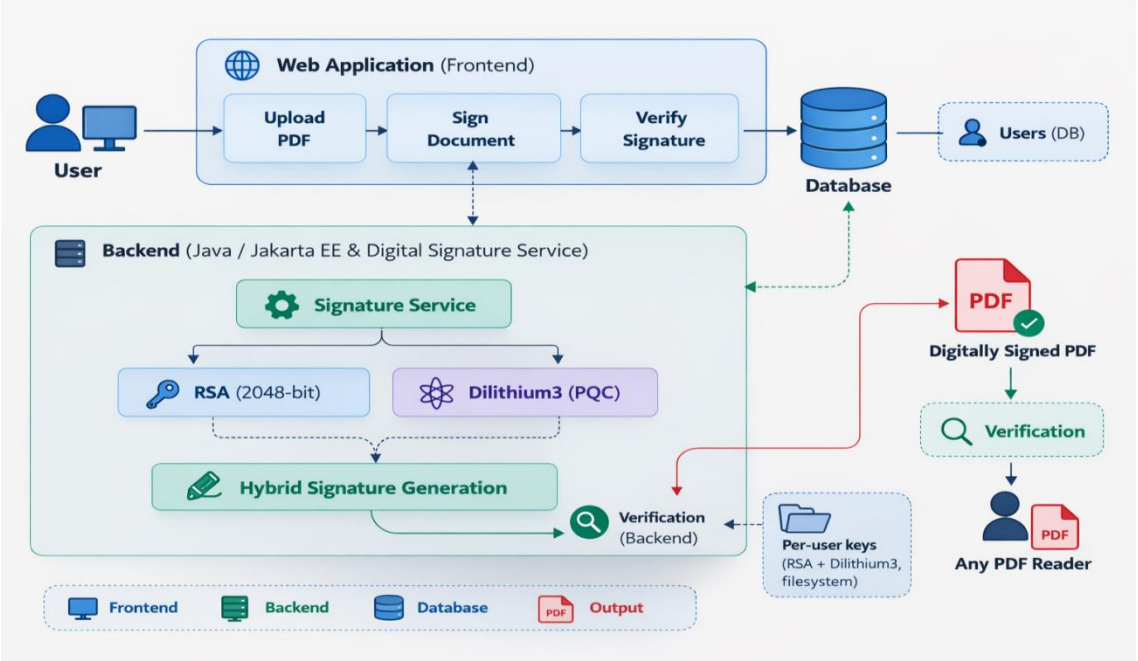
4. Σχεδιασμός Συστήματος

4.1 Αρχιτεκτονική

Το σύστημα σχεδιάστηκε ακολουθώντας την αρχιτεκτονική τριών επιπέδων (three-tier architecture), η οποία αποτελεί μία από τις πιο διαδεδομένες προσεγγίσεις στην ανάπτυξη web εφαρμογών. Η συγκεκριμένη αρχιτεκτονική διαχωρίζει το σύστημα σε επίπεδα, επιτρέποντας καλύτερη οργάνωση, επεκτασιμότητα και συντήρηση του κώδικα. Επιπλέον υποστηρίζει επίσης μηχανισμούς αυθεντικοποίησης χρηστών, συμπεριλαμβανομένης της σύνδεσης μέσω Google OAuth 2.0.Συγκεκριμένα, το σύστημα αποτελείται από τα εξής επίπεδα:

- Το επίπεδο παρουσίασης (frontend), το οποίο υλοποιείται με JSF/PrimeFaces και επιτρέπει την αλληλεπίδραση του χρήστη με την εφαρμογή.
- Το επίπεδο λογικής (backend), στο οποίο υλοποιούνται οι βασικές λειτουργίες υπογραφής και επαλήθευσης.
- Το επίπεδο δεδομένων (database), στο οποίο αποθηκεύονται τα μεταδεδομένα εγγράφων και υπογραφών.

Η αρχιτεκτονική του συστήματος παρουσιάζεται στο ακόλουθο διάγραμμα.



Εικόνα 2. Αρχιτεκτονική του συστήματος δημιουργίας και επαλήθευσης ψηφιακών υπογραφών.

4.2 Μοντέλο δεδομένων

Το μοντέλο δεδομένων του συστήματος σχεδιάστηκε με στόχο την αποθήκευση των απαραίτητων πληροφοριών για τα έγγραφα, τις ψηφιακές υπογραφές και τα γεγονότα (audit events), διασφαλίζοντας παράλληλα την ακεραιότητα και τη συσχέτιση των δεδομένων. Οι οντότητες του συστήματος συνδέονται μεταξύ τους μέσω σχέσεων ένα-προς-πολλά (one-to-many), επιτρέποντας την αποδοτική διαχείριση πολλαπλών υπογραφών και ενεργειών για κάθε έγγραφο.

DocumentFile (document_file): DOCUMENT_ID, FILENAME, CONTENT_TYPE, FILE_SIZE, SHA256, STORAGE_PATH, UPLOAD_TIME, EMAIL_SENT.

DocumentSignature (document_signature): SIGNATURE_ID, DOCUMENT_ID (FK), SCHEME (RSA|DILITHIUM|HYBRID), HASH_ALG, RSA_SIGNATURE, PQC_SIGNATURE, RSA_OK, PQC_OK, OVERALL_OK, SIGN_TIME, idUSER.

TbAuditEvent (tb_audit_event): AUDIT_ID, DOCUMENT_ID, USERNAME, ACTION_TYPE, RESULT_STATUS, IP_ADDRESS, USER_AGENT, CREATED_AT.

Εκτός από τα entities εγγράφων και υπογραφών, το σύστημα χρησιμοποιεί πίνακες διαχείρισης χρηστών και δικαιωμάτων πρόσβασης:

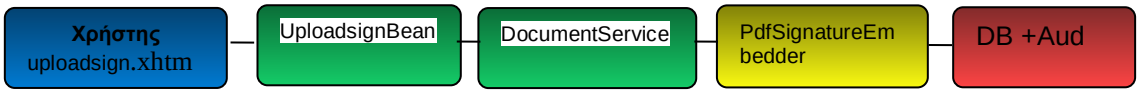
- tbluser (χρήστες): idUser (PK), username, password, create_time, userActive, theme, layoutMode, lightMenu, resetPassword, avatar, idRole (FK → tblroles), name, surname, phone, email. Ο πίνακας συνδέεται με το document_signature μέσω του πεδίου idUser (ο υπογράφων ανά έγγραφο) και με τη διαχείριση προσωπικών κλειδιών (φάκελος data/users/ ανά χρήστη).
- tblroles (ρόλοι): idRole (PK), RoleDesc (π.χ. Administrator, SimpleUser, Guest).
- tblrolerights (δικαιώματα ανά ρόλο): σύνθετο κλειδί (idRole, idPage, idElement).Συνδέει κάθε ρόλο με σελίδες και στοιχεία διεπαφής (FK → tblroles, tblpages, tblelements) για έλεγχο πρόσβασης ανά σελίδα και λειτουργία (R9).

Οι πίνακες document_file, document_signature και tb_audit_event αποτελούν τον πυρήνα της λειτουργίας υπογραφών· οι tbluser, tblroles, tblrolerights, tblpages και tblelements υποστηρίζουν την αυθεντικοποίηση και τον έλεγχο πρόσβασης ανά σελίδα.

4.3 Ροές λειτουργίας του συστήματος

Στην ενότητα αυτή παρουσιάζονται οι βασικές ροές λειτουργίας του συστήματος, οι οποίες περιγράφουν τα βήματα που ακολουθούνται κατά την αλληλεπίδραση του χρήστη με την εφαρμογή.

- **Upload & Sign:** Χρήστης → uploadsign.xhtml → UploadsignBean → DocumentService.signDocument() → PdfSignatureEmbedder → FileStorageService, DB, DocumentAuditService.
- **Verify:** Χρήστης → verify.xhtml → VerifyBean → DocumentService.verify() → PdfSignatureVerifier (ByteRange) → αποτέλεσμα επαλήθευσης.
- **Audit:** Λίστα εγγράφων → επιλογή → επαλήθευση / email / download → ιστορικό events ανά έγγραφο.



Οι κύριες λειτουργίες περιλαμβάνουν την αποστολή και υπογραφή εγγράφων PDF(Upload & Sign) καθώς και την επαλήθευση των ψηφιακών υπογραφών (Verify).

4.3.1 Ροή Upload & Sign

Η διαδικασία «ανέβασμα και υπογραφή» αποτελεί μία από τις βασικές λειτουργίες του συστήματος, επιτρέποντας στον χρήστη να ανεβάσει ένα έγγραφο PDF και να το υπογράψει ψηφιακά με το επιθυμητό σχήμα υπογραφής.

Η ροή «ανέβασμα και υπογραφή» (Upload & Sign) ξεκινά από τη σελίδα uploadsign.xhtml του frontend, η οποία υλοποιείται με τη βιβλιοθήκη PrimeFaces. Ο χρήστης επιλέγει ένα αρχείο PDF προς αποστολή και στη συνέχεια επιλέγει το σχήμα υπογραφής που επιθυμεί να χρησιμοποιήσει: RSA, Dilithium3 (PQC) ή υβριδικό RSA+Dilithium.

Η λογική της σελίδας υλοποιείται από το JSF bean UploadsignBean, το οποίο έχει scope @ViewScoped. Η κύρια μέθοδος της ροής είναι η uploadAndSign(). Η μέθοδος αυτή εκτελεί τα βασικά βήματα της διαδικασίας:

- Έλεγχο εγκυρότητας του αρχείου ώστε να διασφαλιστεί ότι έχει επιλεγεί μη κενό αρχείο PDF.
- Υπολογισμό του SHA-256 hash του αρχείου μέσω της κλάσης FileStorageService.
- Δημιουργία και αποθήκευση ενός αντικειμένου DocumentFile στη βάση δεδομένων, το οποίο περιλαμβάνει τα μεταδεδομένα του εγγράφου, όπως όνομα αρχείου, μέγεθος, τύπο περιεχομένου, hash και χρόνο αποστολής.
- Αποθήκευση των πραγματικών bytes του PDF στο filesystem σε σταθερή διαδρομή της μορφής:

uploads/{documentId}.pdf

Στο στάδιο αυτό ενημερώνεται το πεδίο storagePath του αντικειμένου DocumentFile.

Τέλος, το αντικείμενο DocumentFile επαναφορτώνεται ως managed JPA entity, ώστε να χρησιμοποιηθεί με ασφάλεια στο επόμενο στάδιο της διαδικασίας υπογραφής.

4.3.2 Δημιουργία ψηφιακής υπογραφής

Η δημιουργία της ψηφιακής υπογραφής αποτελεί το βασικό στάδιο της διαδικασίας, στο οποίο εφαρμόζονται οι κρυπτογραφικοί αλγόριθμοι για την παραγωγή της υπογραφής. Η κεντρική υπηρεσία που υλοποιεί τη διαδικασία υπογραφής είναι η κλάση *DocumentService*, και συγκεκριμένα η μέθοδος:

- *signDocument(DocumentFile doc, String scheme, Tbluser user, char[] keystorePassword)*

Η μέθοδος αυτή δημιουργεί ένα νέο αντικείμενο DocumentSignature, το οποίο συνδέεται με το αντίστοιχο DocumentFile. Στο αντικείμενο αυτό αποθηκεύονται:

- το σχήμα υπογραφής (RSA, DILITHIUM ή HYBRID)
- ο χρήστης που πραγματοποίησε την υπογραφή
- η χρονική στιγμή υπογραφής

Στη συνέχεια καλείται η κλάση *PdfSignatureEmbedder*, η οποία είναι υπεύθυνη για την κρυπτογραφική υπογραφή του PDF και την ενσωμάτωση της υπογραφής στο αρχείο.

Μετά την ολοκλήρωση της διαδικασίας, το *UploadsignBean* αποθηκεύει το αντικείμενο *DocumentSignature* στη βάση δεδομένων και καταγράφει μέσω της υπηρεσίας *DocumentAuditService* δύο γεγονότα audit:

- UPLOAD
- SIGN

μαζί με το αποτέλεσμα της ενέργειας.

4.3.3 Ενσωμάτωση υπογραφής στο PDF

Η ενσωμάτωση της υπογραφής στο PDF πραγματοποιείται με τρόπο που να διατηρεί τη συμβατότητα με τα υπάρχοντα πρότυπα και εργαλεία ανάγνωσης εγγράφων. Η κλάση *PdfSignatureEmbedder* χρησιμοποιεί τη βιβλιοθήκη Apache PDFBox για να ανοίξει το αρχικό PDF και να προσθέσει ένα PDF

signature dictionary (PDSignature).

Στο dictionary αυτό ορίζονται βασικές παράμετροι της υπογραφής, όπως:

- Filter
- SubFilter
- Name
- Reason
- Location

Για την οπτική αναπαράσταση της υπογραφής (visible signature) χρησιμοποιείται η κλάση *PdfVisualSignatureUtil*, η οποία δημιουργεί ένα μικρό template PDF μέσω της μεθόδου:=

createVisualSignatureTemplate()

Το template αυτό περιλαμβάνει ένα signature field και ένα ορατό πλαίσιο που εμφανίζει πληροφορίες όπως:

- το κείμενο «DIGITAL SIGNATURE»
- το όνομα του υπογράφοντα
- το σχήμα υπογραφής που χρησιμοποιήθηκε

Στη συνέχεια η PdfSignatureEmbedder χρησιμοποιεί τη λειτουργία external signing της PDFBox μέσω της μεθόδου:

saveIncrementalForExternalSigning

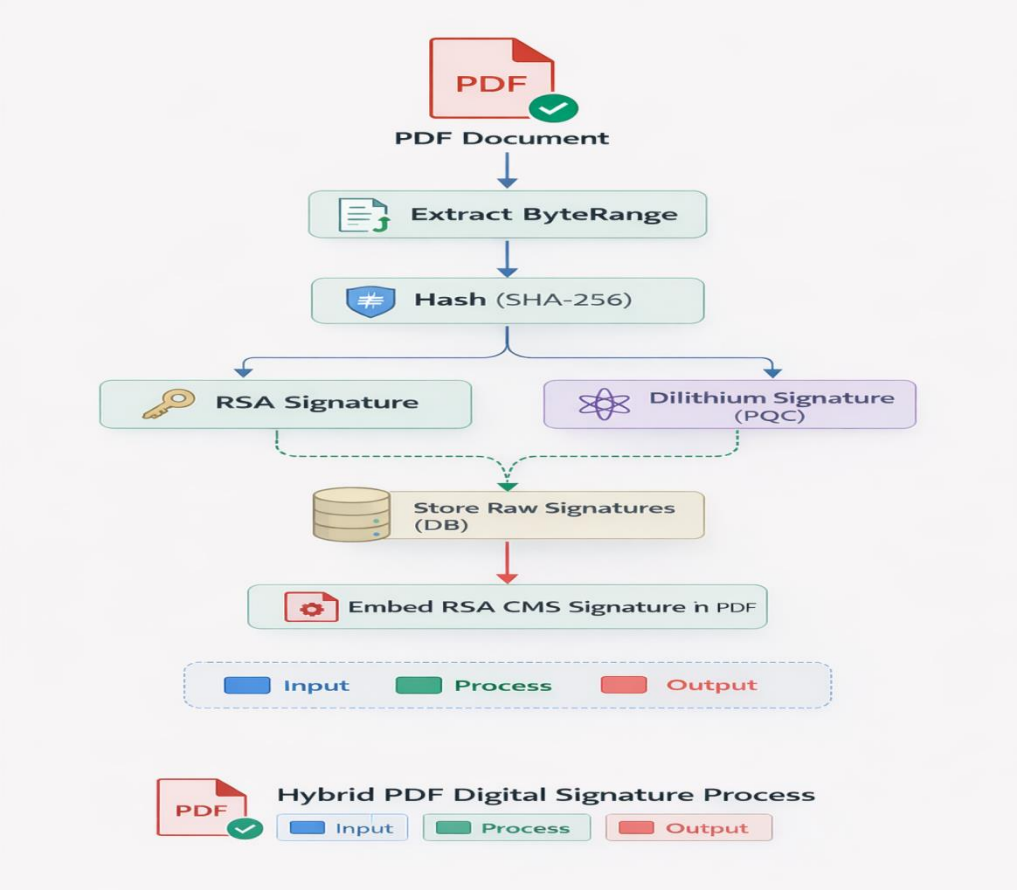
Η βιβλιοθήκη υπολογίζει το ByteRange του PDF και επιστρέφει τα ακριβή bytes που πρέπει να υπογραφούν μέσω:

ExternalSigningSupport.getContent().readAllBytes()

Τα bytes αυτά αποτελούν τα signed bytes του εγγράφου.

4.3.4 Υποστήριξη RSA, Dilithium και υβριδικών υπογραφών

Το σύστημα υποστηρίζει πολλαπλά σχήματα υπογραφής, επιτρέποντας τη χρήση τόσο κλασικών όσο και μετα-κβαντικών αλγορίθμων. Συγκεκριμένα, χρησιμοποιείται ο ευρέως διαδεδομένος αλγόριθμος RSA για συμβατότητα με τα υπάρχοντα συστήματα και τους PDF readers, καθώς και ο μετα-κβαντικός αλγόριθμος Dilithium3, ο οποίος έχει επιλεγεί από το NIST ως πρότυπο για μετα-κβαντικές ψηφιακές υπογραφές. Η ταυτόχρονη χρήση των δύο αλγορίθμων επιτρέπει την υλοποίηση ενός υβριδικού σχήματος υπογραφής (hybrid signature scheme), το οποίο παρέχει συμβατότητα με τα σημερινά πρότυπα υπογραφής εγγράφων, ενώ παράλληλα προσφέρει ανθεκτικότητα απέναντι σε μελλοντικές επιθέσεις από κβαντικούς υπολογιστές. Με τον τρόπο αυτό το σύστημα διατηρεί διαλειτουργικότητα με τα υπάρχοντα εργαλεία επαλήθευσης PDF, ενώ παράλληλα ενσωματώνει τεχνολογίες που θεωρούνται κατάλληλες για την εποχή της μετα-κβαντικής κρυπτογραφίας.



Εικόνα 3. Διαδικασία δημιουργίας υβριδικής ψηφιακής υπογραφής PDF με χρήση RSA και Dilithium.

Τα signed bytes που προκύπτουν από το ByteRange του PDF χρησιμοποιούνται για τη δημιουργία των ψηφιακών υπογραφών.

Αρχικά, τα δεδομένα υπογράφονται με χρήση των αλγορίθμων RSA και/ή Dilithium3, ανάλογα με το σχήμα υπογραφής που έχει επιλέξει ο χρήστης. Τα αποτελέσματα αποθηκεύονται ως raw υπογραφές στο αντικείμενο DocumentSignature της βάσης δεδομένων, στα πεδία:

- rsaSignature
- pqcSignature

Στη συνέχεια, τα ίδια δεδομένα υπογράφονται εκ νέου με RSA σε μορφή CMS/PKCS#7, μέσω της κλάσης *RsaSignatureInterface*. Το παραγόμενο CMS blob τοποθετείται στο πεδίο /Contents του PDF, όπως απαιτεί το πρότυπο PAdES, ώστε το υπογεγραμμένο έγγραφο να είναι συμβατό με τους υπάρχοντες PDF readers.

Η υλοποίηση υποστηρίζει τρία σχήματα υπογραφής:

- **RSA μόνο**

Η raw υπογραφή και το CMS βασίζονται στον αλγόριθμο SHA256withRSA.

- **Dilithium3 μόνο**

Στο σχήμα Dilithium3 μόνο, η κύρια υπογραφή που αποθηκεύεται και επαληθεύεται από την εφαρμογή είναι η μετα-κβαντική υπογραφή Dilithium3. Ωστόσο, για λόγους συμβατότητας με το πρότυπο PDF/PAdES και τους υπάρχοντες PDF readers, το πεδίο /Contents του PDF εξακολουθεί να περιέχει υπογραφή τύπου CMS βασισμένη σε RSA. Η επαλήθευση της μετα-κβαντικής υπογραφής πραγματοποιείται από την υπηρεσία επαλήθευσης της εφαρμογής.

- **Υβριδικό RSA + Dilithium**

Στο υβριδικό σχήμα παράγονται και οι δύο υπογραφές πάνω στα ίδια ByteRange bytes του εγγράφου. Οι υπογραφές αποθηκεύονται στο αντικείμενο DocumentSignature, επιτρέποντας τόσο συμβατότητα με τα υπάρχοντα συστήματα όσο και υποστήριξη μετα-κβαντικής ασφάλειας.

Μετά την ολοκλήρωση της διαδικασίας, το παραγόμενο υπογεγραμμένο αρχείο αντικαθιστά το αρχικό PDF στο filesystem.

4.3.5 Διαχείριση κρυπτογραφικών κλειδιών

Η διαχείριση των κρυπτογραφικών κλειδιών αποτελεί κρίσιμο στοιχείο για την ασφάλεια του συστήματος. Τα κρυπτογραφικά κλειδιά δεν αποθηκεύονται στη βάση δεδομένων αλλά στο filesystem του server. Για κάθε χρήστη δημιουργείται ένας φάκελος της μορφής:

data/users/<username>/

Ο φάκελος αυτός περιέχει:

- ένα PKCS#12 keystore με το ιδιωτικό RSA κλειδί και το πιστοποιητικό
- το αρχείο rsa-public.key
- το αρχείο pqc-private.key
- το αρχείο pqc-public.key

Τα αρχεία αυτά είναι αποθηκευμένα σε μορφή Base64 DER.

Η κλάση *UserKeystoreService* είναι υπεύθυνη για τη δημιουργία και διαγραφή των προσωπικών κλειδιών των χρηστών, ενώ η κλάση KeyLoader αναλαμβάνει τη φόρτωσή τους κατά τη διαδικασία υπογραφής και επαλήθευσης.

4.4 Ροή Verify (Επαλήθευση υπογραφής)

Η διαδικασία επαλήθευσης επιτρέπει τον έλεγχο της εγκυρότητας μιας ψηφιακής υπογραφής, διασφαλίζοντας ότι το έγγραφο δεν έχει τροποποιηθεί και ότι προέρχεται από τον υπογράφοντα.

4.4.1 Επιλογή εγγράφου

Η διαδικασία επαλήθευσης ξεκινά από τη σελίδα *verify.xhtml*, όπου ο χρήστης μπορεί να δει μια λίστα όλων των εγγράφων που είναι αποθηκευμένα στο σύστημα.

Ο χρήστης επιλέγει ένα συγκεκριμένο documentId και πατά το κουμπί Verify. Η ενέργεια αυτή καλεί τη μέθοδο *verify()* του *VerifyBean*, η οποία με τη σειρά της καλεί τη μέθοδο:

DocumentService.verify(docId)

Η μέθοδος αυτή φορτώνει από τη βάση δεδομένων:

- το αντίστοιχο DocumentFile
- την τελευταία DocumentSignature για το συγκεκριμένο έγγραφο

Η ανάκτηση της υπογραφής γίνεται μέσω named query, ταξινομημένης με βάση το πεδίο signTime.

Στη συνέχεια φορτώνονται τα αντίστοιχα δημόσια κλειδιά RSA και Dilithium, με βάση τον χρήστη που υπέγραψε το έγγραφο (idUser).

4.4.2 Επαλήθευση μέσω ByteRange

Η κύρια διαδικασία επαλήθευσης υλοποιείται από την κλάση *PdfSignatureVerifier*.

Αρχικά η μέθοδος:

getSignatureByteRanges(pdfPath)

ανοίγει το PDF μέσω PDFBox και εντοπίζει όλα τα signature fields (PDSignatureField) που διαθέτουν έγκυρο ByteRange και μη κενό πεδίο /Contents.

Για κάθε ByteRange η μέθοδος:

getSignedBytes(pdfPath, byteRange)

χρησιμοποιεί RandomAccessFile ώστε να διαβάσει τα δύο τμήματα που ορίζει το ByteRange:

[start1, length1]

[start2, length2]

Τα δύο τμήματα ενώνονται σε έναν ενιαίο πίνακα bytes. Αυτά αποτελούν τα signed bytes που είχαν υπογραφεί κατά τη φάση Upload & Sign.

4.4.3 Επαλήθευση RSA και Dilithium

Αν υπάρχει raw RSA υπογραφή στη βάση δεδομένων, δημιουργείται ένα instance:

Signature SHA256withRSA

Το instance ενημερώνεται με τα signed bytes και επαληθεύει το rsaSignature.

Με αντίστοιχο τρόπο, αν υπάρχει PQC υπογραφή, δημιουργείται instance:

Signature DILITHIUM3

μέσω της βιβλιοθήκης BouncyCastle, και επαληθεύεται το pqcSignature.

Σε περίπτωση που το PDF περιέχει πολλαπλά ByteRanges (δηλαδή πολλές υπογραφές), ο αλγόριθμος δοκιμάζει διαδοχικά κάθε ByteRange μέχρι να εντοπιστεί εκείνο για το οποίο όλες οι υπογραφές είναι έγκυρες.

Παράλληλα υπολογίζονται δύο επιπλέον ιδιότητες:

- αν υπάρχουν incremental updates
- αν το ByteRange καλύπτει ολόκληρο το αρχείο

4.4.4 Fallback επαλήθευση μέσω hash

Σε περιπτώσεις όπου το PDF δεν διαθέτει έγκυρο signature field, ενεργοποιείται ένας εναλλακτικός μηχανισμός επαλήθευσης.

Στον μηχανισμό αυτό χρησιμοποιείται το SHA-256 hash που είναι αποθηκευμένο στο DocumentFile.sha256. Το hash μετατρέπεται σε bytes και οι raw υπογραφές από τη βάση δεδομένων επαληθεύονται πάνω σε αυτό.

Η μέθοδος αυτή είναι λιγότερο ακριβής, καθώς δεν ελέγχει πιθανές τροποποιήσεις μετά την υπογραφή. Ωστόσο επιτρέπει την επαλήθευση παλαιότερων ή προβληματικών εγγράφων.

4.4.5 Αποτελέσματα επαλήθευσης

Η μέθοδος *DocumentService.verify()* επιστρέφει ένα αντικείμενο *VerificationResult*, το οποίο περιλαμβάνει:

- rsaOk
- pqcOk
- overall
- hasIncrementalUpdates
- coversWholeFile

καθώς και τα αντίστοιχα entities DocumentFile και DocumentSignature.

Το VerifyBean μετατρέπει τα αποτελέσματα αυτά σε φιλική μορφή για τον χρήστη (π.χ. OK, FAIL, N/A) και τα εμφανίζει σε πίνακα πληροφοριών στη σελίδα επαλήθευσης. Παράλληλα δημιουργούνται τα κατάλληλα audit events για επιτυχημένη ή αποτυχημένη επαλήθευση.

Με τον τρόπο αυτό το backend παρέχει μια πλήρη και τεχνικά συνεπή ροή ανέβασμα – υπογραφή – επαλήθευση, βασισμένη στη δομή υπογραφών PDF (ByteRange, /Contents, CMS) και επεκταμένη ώστε να υποστηρίζει υβριδικές υπογραφές με RSA και Dilithium3. Συνολικά, η διαδικασία επαλήθευσης καλύπτει τόσο τον έλεγχο της εγκυρότητας της υπογραφής όσο και τον εντοπισμό πιθανών

μεταγενέστερων τροποποιήσεων του εγγράφου.

Ο σχεδιασμός του συστήματος βασίζεται σε σύγχρονες αρχές ανάπτυξης λογισμικού, συνδυάζοντας την επεκτασιμότητα, την ασφάλεια και τη διαλειτουργικότητα. Οι επιλογές αρχιτεκτονικής και υλοποίησης επιτρέπουν την υποστήριξη τόσο κλασικών όσο και μετα-κβαντικών τεχνολογιών, προετοιμάζοντας το σύστημα για μελλοντικές εξελίξεις στον τομέα της κρυπτογραφίας.

5. Υλοποίηση

5.1 Τεχνολογίες

Για την υλοποίηση του συστήματος χρησιμοποιήθηκε ένα σύνολο τεχνολογιών που καλύπτουν τις ανάγκες του frontend, του backend, της διαχείρισης δεδομένων και της κρυπτογραφίας.

- **Η Java 21** χρησιμοποιήθηκε ως βασική γλώσσα προγραμματισμού για την ανάπτυξη του συστήματος, λόγω της σταθερότητας, της απόδοσης και της υποστήριξης σύγχρονων χαρακτηριστικών.
- Το **Jakarta EE / JSF** αξιοποιήθηκε για την ανάπτυξη της web εφαρμογής και τη διαχείριση της επιχειρησιακής λογικής και της διεπαφής χρήστη.
- **Το PrimeFaces** χρησιμοποιήθηκε για την ανάπτυξη διαδραστικών στοιχείων διεπαφής χρήστη στο frontend.
- **Ο WildFly** επιλέχθηκε ως application server για την εκτέλεση της Jakarta EE εφαρμογής.
- **Τα JPA / Hibernate** χρησιμοποιήθηκαν για την αντιστοίχιση αντικειμένων με τη σχεσιακή βάση δεδομένων και τη διαχείριση των οντοτήτων του συστήματος.
- **Η MySQL 8** χρησιμοποιήθηκε ως σύστημα διαχείρισης βάσης δεδομένων για την αποθήκευση μεταδεδομένων εγγράφων, υπογραφών και ενεργειών audit.
- **Το Apache PDFBox** χρησιμοποιήθηκε για την ανάγνωση, τροποποίηση και υπογραφή αρχείων PDF, καθώς και για τη διαχείριση του ByteRange και των signature fields.
- **Η βιβλιοθήκη Bouncy Castle** χρησιμοποιήθηκε για την υλοποίηση των κρυπτογραφικών λειτουργιών, τόσο για RSA όσο και για Dilithium3 καθώς και για τη δημιουργία ζευγών κλειδιών μέσω μηχανισμών όπως KeyPairGenerator και τη διαχείριση keystores (PKCS#12).
- **Η βιβλιοθήκη Simple Java Mail** χρησιμοποιήθηκε για την αποστολή υπογεγραμμένων εγγράφων μέσω email.

5.2 Βασικές λειτουργίες

Στην παρούσα ενότητα παρουσιάζονται οι βασικές λειτουργίες που υλοποιήθηκαν στο σύστημα, όπως η δημιουργία ψηφιακής υπογραφής, η επαλήθευση υπογραφών, η καταγραφή ενεργειών και η διαχείριση κρυπτογραφικών κλειδιών.

- **Υπογραφή:** Το component PdfSignatureEmbedder αναλαμβάνει την ενσωμάτωση της ψηφιακής υπογραφής στο PDF. Η διαδικασία περιλαμβάνει την ανάγνωση των ByteRange bytes από το έγγραφο PDF, την παραγωγή ψηφιακής υπογραφής με χρήση των αλγορίθμων RSA και/ή Dilithium, την αποθήκευση των raw υπογραφών στη βάση δεδομένων και την ενσωμάτωση του CMS signature (RSA) στο πεδίο /Contents του PDF για συμβατότητα με συμβατικούς PDF readers.
- **Επαλήθευση:** Η επαλήθευση των υπογραφών πραγματοποιείται από το component PdfSignatureVerifier, το οποίο διαβάζει το ByteRange από το PDF, ανακτά τα υπογεγραμμένα bytes και επαληθεύει τις raw υπογραφές που έχουν αποθηκευτεί στη βάση δεδομένων. Παράλληλα ελέγχεται εάν υπάρχουν incremental updates στο PDF και εάν η υπογραφή καλύπτει το σύνολο του αρχείου.
- **Audit:** Η υπηρεσία DocumentAuditService καταγράφει κάθε ενέργεια που πραγματοποιείται στο σύστημα, όπως UPLOAD, SIGN, VERIFY, EMAIL_SENT και DOWNLOAD, αποθηκεύοντας πληροφορίες όπως documentId, username, IP address, User-Agent και status (SUCCESS / FAILURE).
- **Διαχείριση Κλειδιών**: Το σύστημα υποστηρίζει μηχανισμό διαχείρισης κρυπτογραφικών κλειδιών ανά χρήστη. Κάθε χρήστης διαθέτει το δικό του ζεύγος κλειδιών (δημόσιο και ιδιωτικό) για τους αλγορίθμους RSA και Dilithium3. Τα κλειδιά δημιουργούνται μέσω ειδικής σελίδας της εφαρμογής και αποθηκεύονται με ασφαλή τρόπο στο σύστημα.

Κατά τη δημιουργία και χρήση των κλειδιών, ο χρήστης εισάγει έναν μυστικό κωδικό (password), ο οποίος χρησιμοποιείται για την προστασία του keystore και το ξεκλείδωμα των ιδιωτικών κλειδιών κατά τη διαδικασία υπογραφής. Ο κωδικός αυτός δεν αποθηκεύεται στο σύστημα, ενισχύοντας την ασφάλεια των κρυπτογραφικών δεδομένων.

Μέσω της ίδιας διεπαφής ο χρήστης μπορεί να δημιουργήσει νέο ζεύγος κλειδιών ή να ανανεώσει υπάρχοντα κλειδιά. Ο μηχανισμός αυτός επιτρέπει την ανεξάρτητη υπογραφή εγγράφων από κάθε χρήστη και υποστηρίζει την ασφαλή διαχείριση των ψηφιακών υπογραφών στο σύστημα. Το ιδιωτικό κλειδί χρησιμοποιείται για τη δημιουργία της ψηφιακής υπογραφής, ενώ το αντίστοιχο δημόσιο κλειδί χρησιμοποιείται κατά τη διαδικασία επαλήθευσης.

- **Υποστήριξη αυθεντικοποίησης μέσω Google:** Το σύστημα υποστηρίζει μηχανισμό

αυθεντικοποίησης χρηστών μέσω Google, αξιοποιώντας το πρωτόκολλο OAuth 2.0. Μέσω της λειτουργίας αυτής, οι χρήστες μπορούν να συνδεθούν στην εφαρμογή χρησιμοποιώντας τον λογαριασμό τους Google, χωρίς την ανάγκη δημιουργίας νέων διαπιστευτηρίων. Κατά τη διαδικασία αυθεντικοποίησης, το σύστημα λαμβάνει βασικά στοιχεία του χρήστη (όπως email και όνομα) και δημιουργεί ή ενημερώνει την αντίστοιχη εγγραφή στη βάση δεδομένων. Με τον τρόπο αυτό επιτυγχάνεται απλοποίηση της διαδικασίας εισόδου, καθώς και βελτίωση της εμπειρίας χρήστη.

5.3 Δομή κώδικα

Η δομή του κώδικα οργανώθηκε σε επιμέρους επίπεδα και κλάσεις, σύμφωνα με τις αρχές διαχωρισμού ευθυνών (separation of concerns). Οι βασικές λειτουργίες της εφαρμογής κατανέμονται σε beans διεπαφής χρήστη, service classes, κλάσεις διαχείρισης PDF και υπογραφών, καθώς και οντότητες για την αλληλεπίδραση με τη βάση δεδομένων.

Ενδεικτικά, οι κλάσεις *UploadsignBean* και *VerifyBean* διαχειρίζονται την αλληλεπίδραση με τον χρήστη, η *DocumentService* υλοποιεί τον βασικό επιχειρησιακό χειρισμό εγγράφων και υπογραφών, ενώ οι *PdfSignatureEmbedder* και *PdfSignatureVerifier* αναλαμβάνουν την ενσωμάτωση και την επαλήθευση υπογραφών στα αρχεία PDF.

Η συγκεκριμένη οργάνωση συμβάλλει στη βελτίωση της αναγνωσιμότητας του κώδικα, στη διευκόλυνση της συντήρησης και στην επεκτασιμότητα του συστήματος.

5.4 Εργαλεία Ανάπτυξης και Χρήση Εργαλείων Τεχνητής Νοημοσύνης

Στην ενότητα αυτή παρουσιάζονται τα βασικά εργαλεία ανάπτυξης που χρησιμοποιήθηκαν για την υλοποίηση και εκτέλεση της εφαρμογής.

Για την υλοποίηση της εφαρμογής χρησιμοποιήθηκε ένα σύνολο σύγχρονων εργαλείων ανάπτυξης λογισμικού που υποστηρίζουν την ανάπτυξη web εφαρμογών σε περιβάλλον Java. Η βασική γλώσσα προγραμματισμού του συστήματος είναι η Java, ενώ το backend της εφαρμογής υλοποιήθηκε με τεχνολογίες Jakarta EE.

Η εφαρμογή εκτελείται σε WildFly Application Server, ο οποίος παρέχει το απαραίτητο περιβάλλον εκτέλεσης για Jakarta EE εφαρμογές και υποστηρίζει υπηρεσίες όπως dependency injection, διαχείριση συναλλαγών και ασφάλεια εφαρμογών. Για τη διαχείριση των εξαρτήσεων και τη διαδικασία build χρησιμοποιήθηκε το εργαλείο Apache Maven.

Για την ανάπτυξη της διεπαφής χρήστη χρησιμοποιήθηκε το framework PrimeFaces, το οποίο βασίζεται στο JavaServer Faces (JSF) και παρέχει έτοιμα components για την υλοποίηση δυναμικών web διεπαφών.

Για τις λειτουργίες κρυπτογραφίας αξιοποιήθηκε η βιβλιοθήκη BouncyCastle, η οποία παρέχει υλοποιήσεις σύγχρονων κρυπτογραφικών αλγορίθμων. Στην εφαρμογή χρησιμοποιήθηκε τόσο ο αλγόριθμος RSA όσο και ο μετα-κβαντικός αλγόριθμος CRYSTALS-Dilithium, με σκοπό την υλοποίηση υπογραφών εγγράφων PDF.

Για την επεξεργασία και την ενσωμάτωση ψηφιακών υπογραφών σε αρχεία PDF χρησιμοποιήθηκε η βιβλιοθήκη Apache PDFBox, η οποία επιτρέπει την ανάγνωση, τροποποίηση και δημιουργία PDF εγγράφων καθώς και την υλοποίηση μηχανισμών ψηφιακής υπογραφής.

Η εφαρμογή χρησιμοποιεί επίσης βάση δεδομένων MySQL, στην οποία αποθηκεύονται

πληροφορίες χρηστών, μεταδεδομένα εγγράφων και στοιχεία που σχετίζονται με τις υπογραφές.

Κατά την ανάπτυξη της εφαρμογής χρησιμοποιήθηκαν και εργαλεία τεχνητής νοημοσύνης ως υποστηρικτικό μέσο προγραμματισμού και τεκμηρίωσης. Η χρήση τους αφορούσε κυρίως την αναζήτηση παραδειγμάτων, τη διατύπωση εναλλακτικών λύσεων, τη διόρθωση σφαλμάτων και την υποβοήθηση στη συγγραφή τεκμηρίωσης. Ωστόσο, η αρχιτεκτονική του συστήματος, οι επιλογές σχεδιασμού, η επιλογή των αλγορίθμων και η τελική αξιολόγηση των λύσεων πραγματοποιήθηκαν από τον ερευνητή. Η χρήση των εργαλείων αυτών λειτούργησε υποστηρικτικά στη διαδικασία ανάπτυξης, χωρίς να αντικαθιστά τη μελέτη, την ανάλυση και την τεχνική κρίση του ερευνητή.

6. Αποτελέσματα και Δοκιμές

6.1 Περιβάλλον δοκιμών

Το σύστημα δοκιμάστηκε σε περιβάλλον ανάπτυξης με χρήση τοπικού application server WildFly και βάσης δεδομένων MySQL. Οι δοκιμές πραγματοποιήθηκαν σε υπολογιστικό σύστημα με επεξεργαστή Intel/AMD, μνήμη RAM 32GB και λειτουργικό σύστημα Windows/Linux.

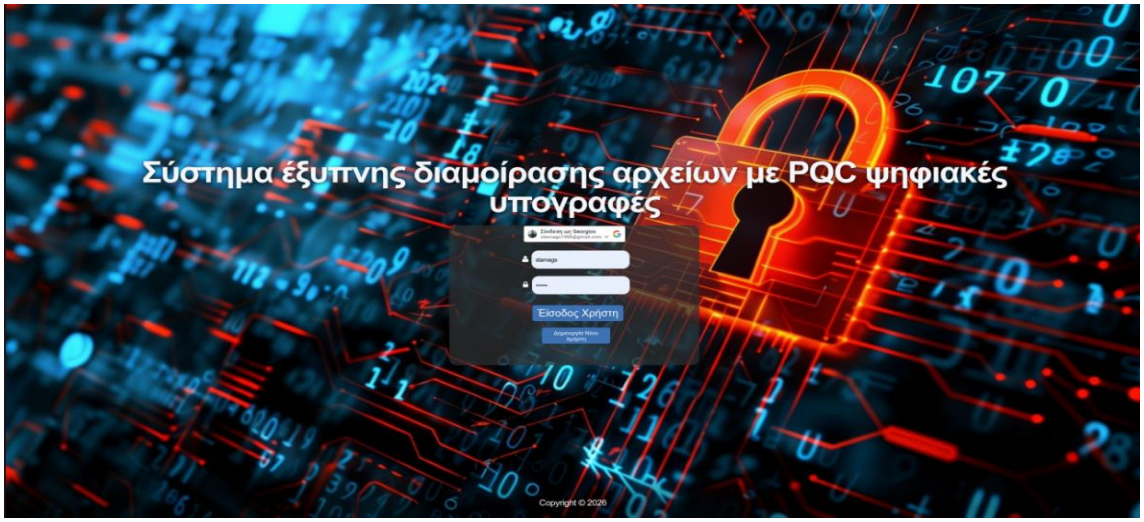
Η εφαρμογή εκτελέστηκε σε web browser (Chrome), ενώ για την επαλήθευση της συμβατότητας των υπογεγραμμένων PDF χρησιμοποιήθηκε το Adobe Acrobat Reader.

6.2 Σενάρια δοκιμών

Για την αξιολόγηση του συστήματος σχεδιάστηκαν και εκτελέστηκαν τα ακόλουθα σενάρια δοκιμών:

- **Αυθεντικοποίηση χρήστη μέσω Google (OAuth 2.0)**. Η λειτουργία αυθεντικοποίησης μέσω Google επιτρέπει την είσοδο χρηστών στο σύστημα χωρίς τη χρήση τοπικών διαπιστευτηρίων. Κατά τη δοκιμή, η διαδικασία σύνδεσης ολοκληρώθηκε επιτυχώς και τα στοιχεία του χρήστη ανακτήθηκαν από την υπηρεσία Google.

Στην παρακάτω εικόνα (εικ. 4) παρουσιάζεται η διαδικασία σύνδεσης μέσω Google authentication.

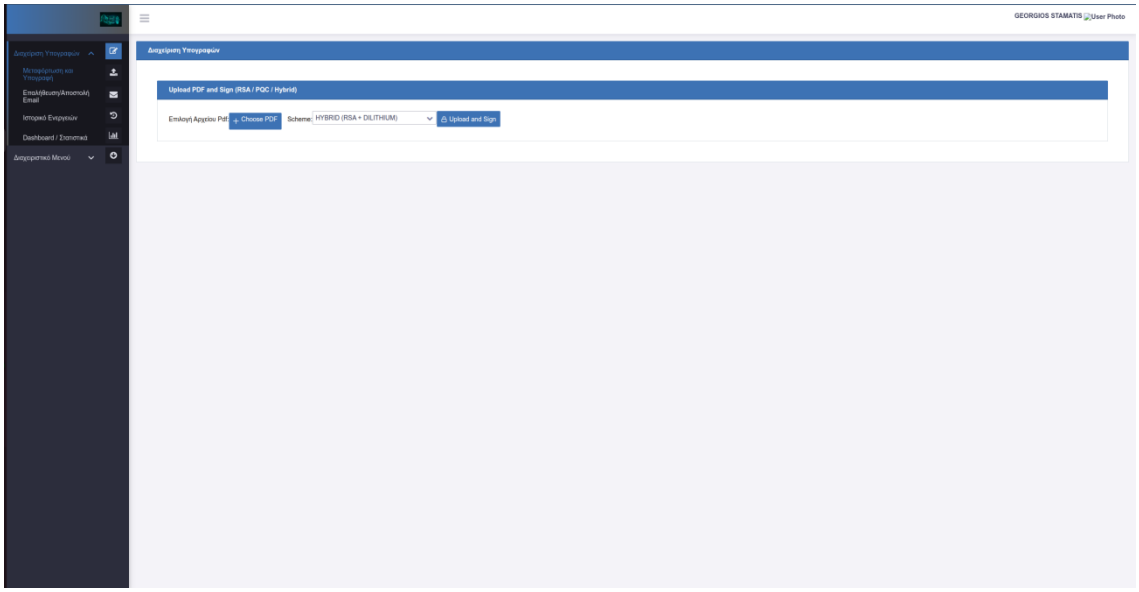


Εικόνα 4. Οθόνη εισόδου χρήστη στο Σύστημα έξυπνης διαμοίρασης αρχείων με PQC ψηφιακές υπογραφές

- **Υπογραφή εγγράφου PDF με χρήση RSA.**Ο χρήστης ανεβάζει ένα αρχείο PDF και επιλέγει το σχήμα υπογραφής RSA. Το σύστημα δημιουργεί την ψηφιακή υπογραφή και την ενσωματώνει στο έγγραφο. Στην Εικόνα 5 παρουσιάζεται η διαδικασία υπογραφής εγγράφου μέσω της εφαρμογής.
- **Υπογραφή εγγράφου PDF με χρήση Dilithium3.** Το σύστημα υποστηρίζει μετα-κβαντική υπογραφή με χρήση του αλγορίθμου Dilithium3. Κατά τη δοκιμή, η υπογραφή δημιουργήθηκε και αποθηκεύτηκε επιτυχώς. Η διαδικασία είναι αντίστοιχη με αυτή που παρουσιάζεται στην Εικόνα 5.
- **Υπογραφή εγγράφου PDF με υβριδικό σχήμα (RSA + Dilithium).** Στο υβριδικό σχήμα δημιουργούνται δύο υπογραφές πάνω στα ίδια δεδομένα (ByteRange). Το σύστημα αποθηκεύει και επαληθεύει και τις δύο υπογραφές. Η ίδια διαδικασία υπογραφής απεικονίζεται στην Εικόνα 5.

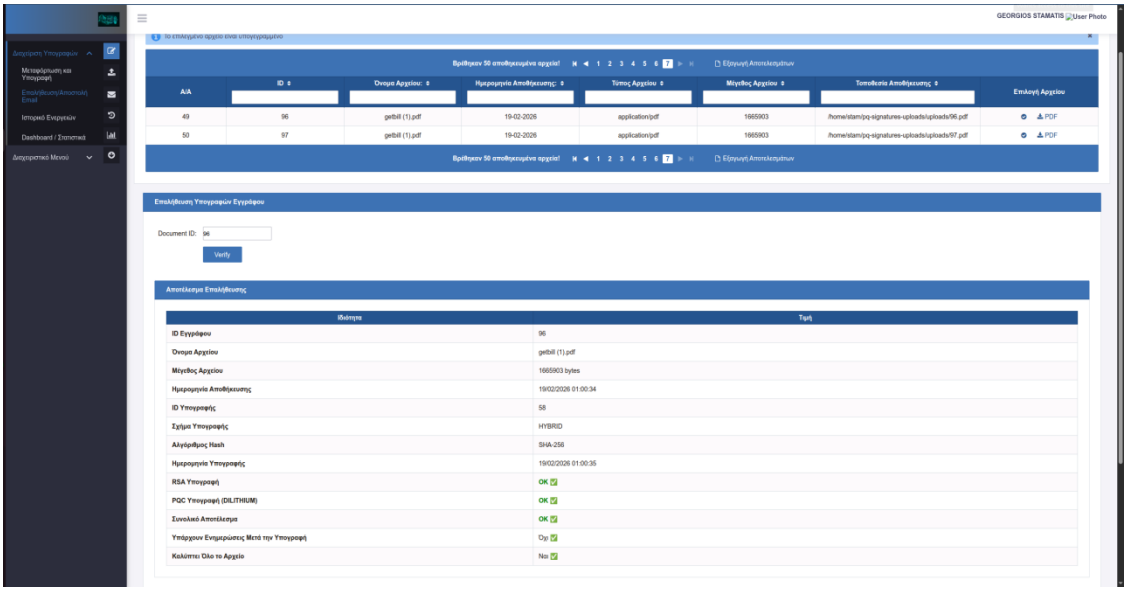
Η ίδια διεπαφή χρησιμοποιείται για όλα τα σχήματα υπογραφής, με μόνη διαφοροποίηση την επιλογή του

αλγορίθμου από τον χρήστη.



Εικόνα 5. Διαδικασία υπογραφής εγγράφου PDF μέσω της εφαρμογής (RSA, Dilithium3 και υβριδικό σχήμα).

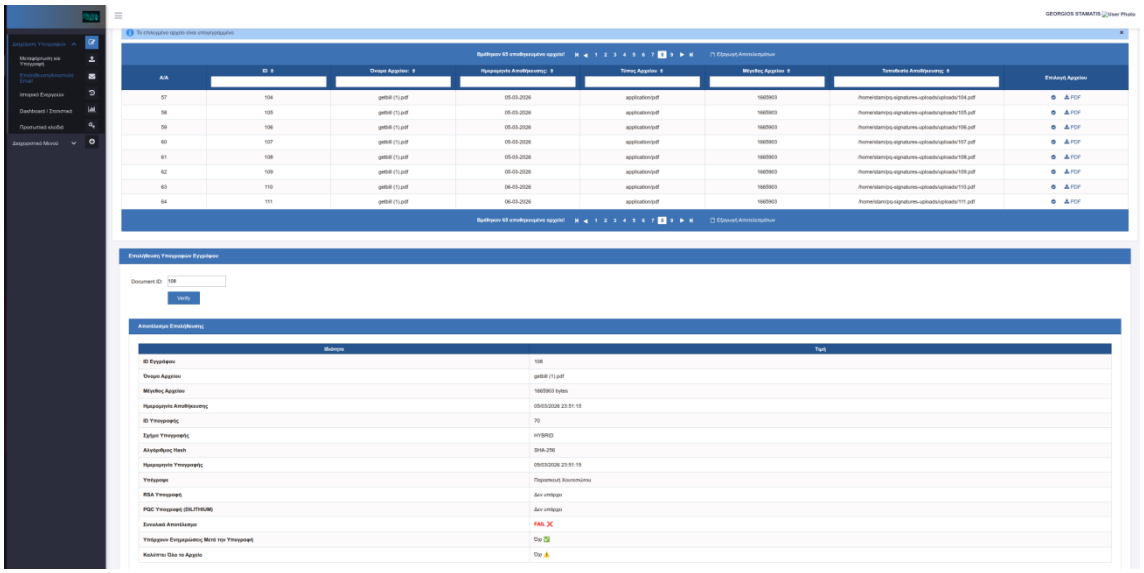
- **Επαλήθευση έγκυρων υπογραφών.** Κατά την επαλήθευση ενός έγκυρου υπογεγραμμένου εγγράφου, το σύστημα επιβεβαιώνει ότι οι υπογραφές είναι έγκυρες και το έγγραφο δεν έχει τροποποιηθεί. Στην παρακάτω εικόνα (εικ. 6) παρουσιάζεται επιτυχής επαλήθευση υπογραφής.



Εικόνα 6. Οθόνη επαλήθευσης – επιλογή εγγράφου και εμφάνιση αποτελεσμάτων (RSA OK, PQC OK, overall).

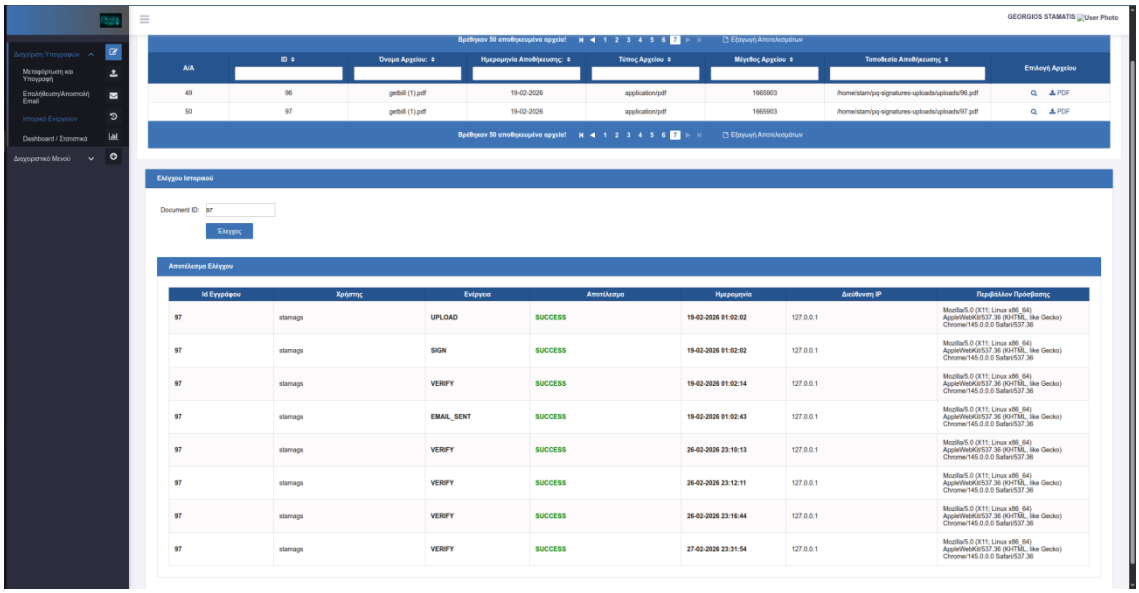
- **Επαλήθευση εγγράφου μετά από τροποποίηση (tampering)**. Σε περίπτωση τροπο-

ποίησης του εγγράφου μετά την υπογραφή, η διαδικασία επαλήθευσης αποτυγχάνει, όπως αναμένεται. Στην παρακάτω εικόνα (εικ. 7) παρουσιάζεται αποτυχημένη επαλήθευση λόγω αλλοίωσης του εγγράφου.



Εικόνα 7. Οθόνη αποτυχημένης επαλήθευσης

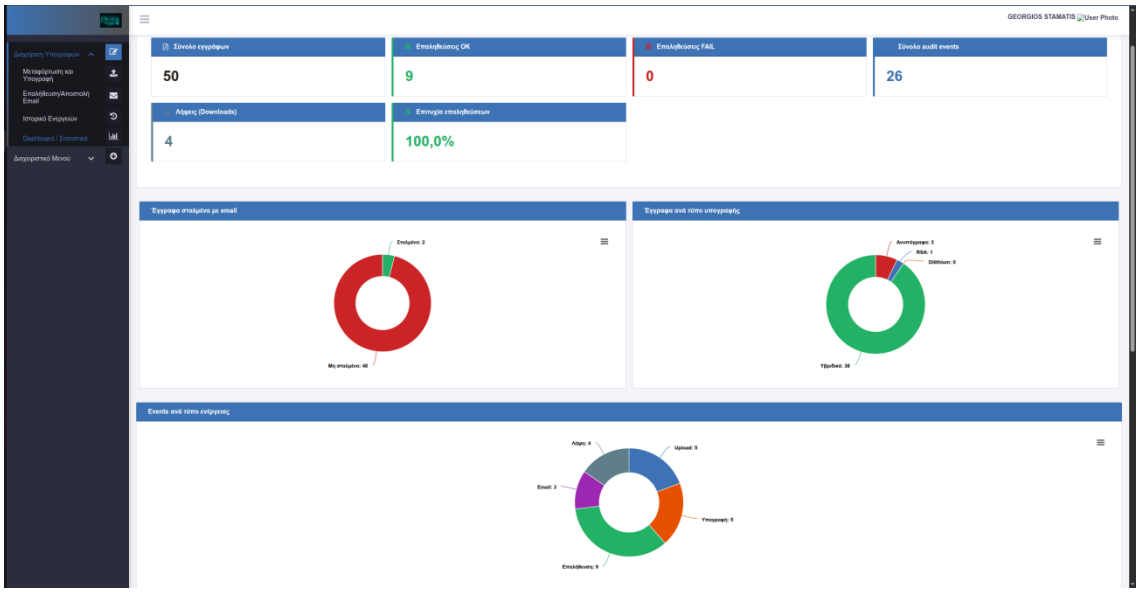
- **Καταγραφή ενεργειών στο audit log.** Κάθε ενέργεια του χρήστη καταγράφεται στο σύστημα audit. Κατά τη δοκιμή, οι ενέργειες (UPLOAD, SIGN, VERIFY, EMAIL_SENT, DOWNLOAD) καταγράφηκαν σωστά. Στην Εικόνα 8 παρουσιάζεται η προβολή των audit events του συστήματος.



Εικόνα 8. Οθόνη ιστορικού ενεργειών – λίστα εγγράφων και καταγραφή ενεργειών ανά έγγραφο.

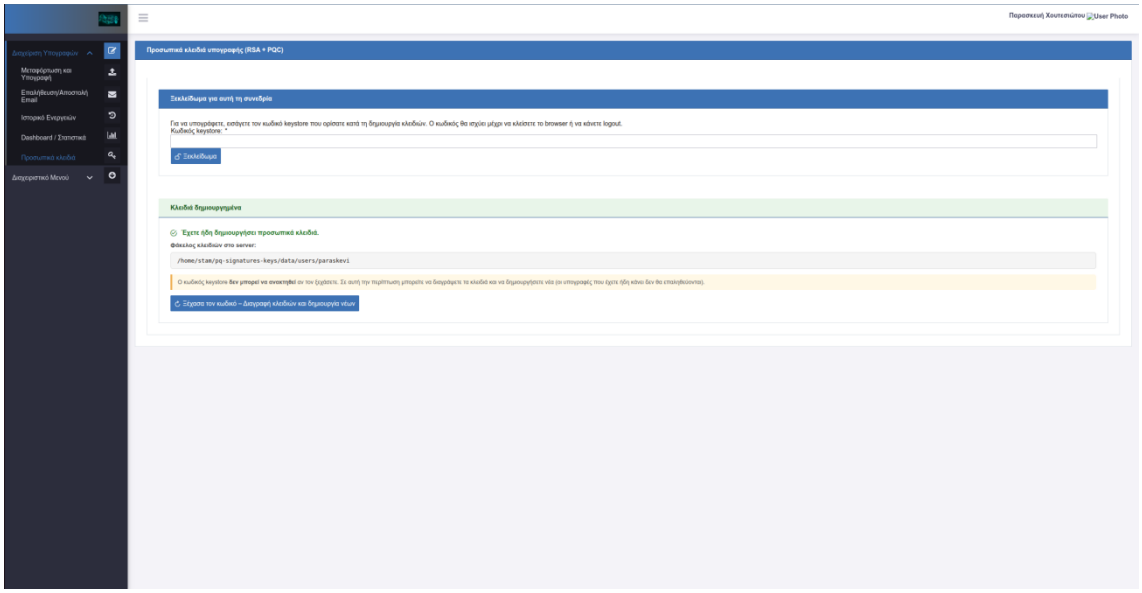
- **Προβολή στατιστικών στοιχείων.** Το σύστημα παρέχει οθόνη προβολής στατιστικών, μέσω της οποίας ο χρήστης μπορεί να δει συγκεντρωτικά στοιχεία σχετικά με τα έγγραφα και τις ενέργειες που έχουν πραγματοποιηθεί (π.χ. πλήθος εγγράφων, υπογραφών, ενεργειών audit και

κατανομές ανά τύπο ενέργειας). Η λειτουργία αυτή συμβάλλει στην καλύτερη κατανόηση της χρήσης του συστήματος και στην παρακολούθηση της δραστηριότητας. Στην παρακάτω εικόνα(εικ. 8) παρουσιάζεται η οθόνη καταγραφής και προβολής στατιστικών του συστήματος.



Εικόνα 9. Οθόνη στατιστικών – σύνολο εγγράφων, επαληθεύσεις και κατανομή τύπων υπογραφής.

- **Διαχείριση προσωπικών κλειδιών χρήστη.** Το σύστημα παρέχει ειδική διεπαφή για τη δημιουργία και το ξεκλείδωμα των προσωπικών κρυπτογραφικών κλειδιών κάθε χρήστη, τόσο για τον αλγόριθμο RSA όσο και για τον μετα-κβαντικό αλγόριθμο Dilithium3. Μέσω της οθόνης αυτής, ο χρήστης μπορεί να δημιουργήσει νέο ζεύγος κλειδιών ή να ξεκλειδώσει το υπάρχον keystore για χρήση κατά τη διαδικασία υπογραφής. Στην Εικόνα 9 παρουσιάζεται η οθόνη δημιουργίας και ξεκλειδώματος προσωπικών κλειδιών (RSA + PQC) ανά χρήστη.



Εικόνα 10. Οθόνη δημιουργίας και ξεκλειδώματος προσωπικών κλειδιών (RSA + PQC) ανά χρήστη.

6.3 Αποτελέσματα

Τα αποτελέσματα των δοκιμών έδειξαν ότι το σύστημα λειτουργεί σωστά σε όλα τα βασικά σενάρια χρήσης. Η υπογραφή εγγράφων με RSA και Dilithium3 ολοκληρώθηκε επιτυχώς, ενώ οι παραγόμενες υπογραφές αποθηκεύτηκαν και επαληθεύτηκαν σωστά από το σύστημα. Στην περίπτωση υβριδικών υπογραφών, και οι δύο υπογραφές δημιουργήθηκαν και επαληθεύτηκαν επιτυχώς, επιβεβαιώνοντας την ορθή λειτουργία του συνδυαστικού σχήματος.

Κατά τη δοκιμή τροποποίησης εγγράφου μετά την υπογραφή, το σύστημα εντόπισε την αλλοίωση, καθώς η επαλήθευση απέτυχε, όπως αναμενόταν. Επιπλέον, τα υπογεγραμμένα έγγραφα άνοιξαν επιτυχώς σε εξωτερικά εργαλεία (Adobe Acrobat Reader), γεγονός που επιβεβαιώνει τη συμβατότητα με το πρότυπο PAdES.

Η λειτουργία αποστολής εγγράφων μέσω email δοκιμάστηκε επιτυχώς, με τα υπογεγραμμένα αρχεία να αποστέλλονται και να παραλαμβάνονται σωστά από τους χρήστες, επιβεβαιώνοντας την ορθή ενσωμάτωση της σχετικής υπηρεσίας.

Τέλος, η διαδικασία δημιουργίας και διαχείρισης προσωπικών κρυπτογραφικών κλειδιών ανά χρήστη λειτούργησε ομαλά, επιτρέποντας τη δημιουργία νέων ζευγών κλειδιών (RSA και Dilithium3), καθώς και το ξεκλείδωμα του keystore για τη χρήση τους κατά την υπογραφή και επαλήθευση εγγράφων.

6.4 Συζήτηση αποτελεσμάτων

Τα αποτελέσματα επιβεβαιώνουν ότι το σύστημα μπορεί να υποστηρίξει με επιτυχία τόσο κλασικές όσο και μετα-κβαντικές ψηφιακές υπογραφές.

Η χρήση υβριδικών υπογραφών αποδείχθηκε ιδιαίτερα σημαντική, καθώς επιτρέπει τη διατήρηση συμβατότητας με τα υπάρχοντα συστήματα, ενώ παράλληλα εισάγει μηχανισμούς ασφάλειας για μελλοντικές απειλές από κβαντικούς υπολογιστές.

Ωστόσο, παρατηρείται ότι οι μετα-κβαντικοί αλγόριθμοι έχουν μεγαλύτερο μέγεθος υπογραφής και ενδέχεται να επηρεάσουν την απόδοση και το μέγεθος των εγγράφων.

Συνολικά, το σύστημα παρουσιάζει υψηλό επίπεδο λειτουργικότητας και ασφάλειας, ενώ μπορεί να αποτελέσει βάση για περαιτέρω βελτιώσεις και επεκτάσεις.

7. Συμπεράσματα

7.1 Αποτελέσματα

Η παρούσα εργασία είχε ως στόχο τη σχεδίαση και υλοποίηση ενός ολοκληρωμένου συστήματος ψηφιακών υπογραφών για έγγραφα PDF, το οποίο να υποστηρίζει τόσο κλασικούς όσο και μετα-κβαντικούς αλγορίθμους.

Τα αποτελέσματα έδειξαν ότι το σύστημα μπορεί να διαχειριστεί αποτελεσματικά τη διαδικασία ανέβασμα–υπογραφή–επαλήθευση εγγράφων, διατηρώντας παράλληλα συμβατότητα με τα υπάρχοντα πρότυπα και εργαλεία. Η υποστήριξη του αλγορίθμου RSA εξασφαλίζει τη διαλειτουργικότητα με τα σημερινά συστήματα, ενώ η ενσωμάτωση του αλγορίθμου Dilithium3 προσφέρει ανθεκτικότητα απέναντι σε μελλοντικές απειλές από κβαντικούς υπολογιστές.

Ιδιαίτερα σημαντική είναι η υλοποίηση υβριδικών υπογραφών, η οποία επιτρέπει τη σταδιακή μετάβαση από την κλασική στην μετα-κβαντική κρυπτογραφία, χωρίς να θυσιάζεται η συμβατότητα με τα υπάρχοντα εργαλεία.

Επιπλέον, το σύστημα ενσωματώνει μηχανισμούς όπως η καταγραφή ενεργειών (audit), η αποστολή εγγράφων μέσω email και η διαχείριση προσωπικών κρυπτογραφικών κλειδιών ανά χρήστη, καθιστώντας το ένα ολοκληρωμένο περιβάλλον διαχείρισης ψηφιακών υπογραφών.

Συνολικά, η εργασία επιβεβαιώνει ότι είναι εφικτή η ενσωμάτωση μετα-κβαντικών αλγορίθμων σε πραγματικά συστήματα, διατηρώντας παράλληλα τη συμβατότητα και τη λειτουργικότητα που απαιτούνται σε σύγχρονες εφαρμογές.

7.2 Μελλοντική Εργασία

Παρόλο που το σύστημα καλύπτει ένα ευρύ φάσμα λειτουργιών, υπάρχουν αρκετές δυνατότητες περαιτέρω βελτίωσης και επέκτασης.

Μία σημαντική κατεύθυνση είναι η υποστήριξη επιπλέον post-quantum αλγορίθμων, καθώς και η αξιολόγηση της απόδοσής τους σε πραγματικές συνθήκες. Επιπλέον, θα μπορούσε να εξεταστεί η ενσωμάτωση μηχανισμών πιστοποίησης μέσω αρχών έκδοσης πιστοποιητικών (Certification Authorities), ώστε να ενισχυθεί η αξιοπιστία των υπογραφών.

Ακόμη, η βελτιστοποίηση της απόδοσης, ιδιαίτερα για μεγάλου μεγέθους έγγραφα και πολλαπλές υπογραφές, αποτελεί ένα σημαντικό πεδίο εξέλιξης. Η χρήση cloud υποδομών ή microservices αρχιτεκτονικής θα μπορούσε επίσης να βελτιώσει την επεκτασιμότητα του συστήματος.

Τέλος, θα μπορούσε να επεκταθεί η εφαρμογή με επιπλέον δυνατότητες, όπως mobile υποστήριξη, προηγμένα dashboards στατιστικών και ενσωμάτωση με εξωτερικά συστήματα διαχείρισης εγγράφων.

8. Βιβλιογραφία

1. NIST, *Post-Quantum Cryptography Standardization*, https://csrc.nist.gov/projects/post-quantum-cryptography
2. NIST FIPS 204, *Module-Lattice-Based Digital Signature Standard*
3. ETSI TS 102 778, *PDF Advanced Electronic Signatures (PAdES)*
4. ISO 32000-2 (PDF 2.0), *Document management – Portable document format*
5. Bouncy Castle Cryptography, https://www.bouncycastle.org/
6. Apache PDFBox, https://pdfbox.apache.org/

Γεώργιος Σταμάτης – Πανεπιστήμιο Πειραιώς /Τμήμα Πληροφορικής – 2026