

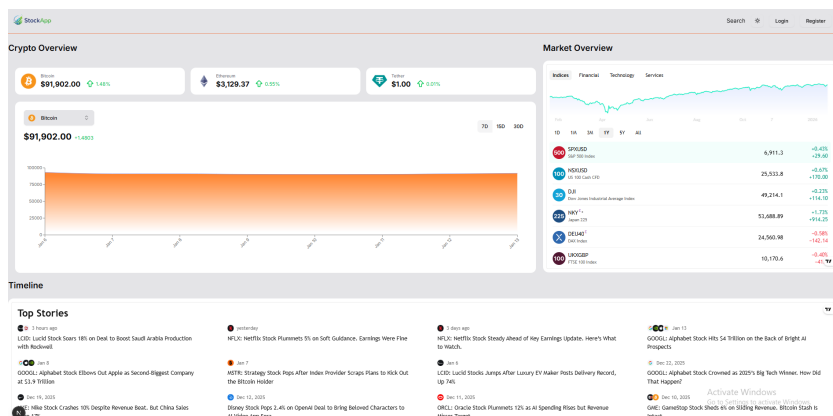


ΔΙΕΘΝΕΣ
ΠΑΝΕΠΙΣΤΗΜΙΟ
ΤΗΣ ΕΛΛΑΔΟΣ

Διεθνές Πανεπιστήμιο Ελλάδος
Τμήμα Μηχανικών Πληροφορικής και Ηλεκτρονικών Συστημάτων

ΠΤΥΧΙΑΚΗ ΕΡΓΑΣΙΑ

AI Financial Dashboard: Εφαρμογή ιστού για ανάλυση και οπτικοποίηση οικονομικών δεδομένων με τεχνικές τεχνητής νοημοσύνης



Φοιτητής:

Θεόδωρος Διαμαντίδης
Αριθμός Μητρώου: 185175

Επιβλέπων:

Στέφανος Ουγιάρογλου

21 Ιανουαρίου 2026

AI Financial Dashboard: Εφαρμογή ιστού για ανάλυση και οπτικοποίηση οικονομικών δεδομένων με τεχνικές τεχνητής νοημοσύνης

Κωδικός Π.Ε.: 25367

Όνοματεπώνυμο φοιτητή: Θεόδωρος Διαμαντίδης

Όνοματεπώνυμο εισηγητή: Στέφανος Ουγιάρογλου

Ημερομηνία ανάληψης: 16-10-2026

Ημερομηνία περάτωσης: 11-02-2026

Βεβαιώνω ότι είμαι ο συγγραφέας αυτής της εργασίας και ότι κάθε βοήθεια την οποία είχα για την προετοιμασία της είναι πλήρως αναγνωρισμένη και αναφέρεται στην εργασία. Επίσης, έχω καταγράψει τις όποιες πηγές από τις οποίες έκανα χρήση δεδομένων, ιδεών, εικόνων και κειμένων, είτε αυτές αναφέρονται ακριβώς είτε παραφρασμένες. Επιπλέον, βεβαιώνω ότι αυτή η εργασία προετοιμάστηκε από εμένα προσωπικά, ειδικά ως πτυχιακή εργασία, στο Τμήμα Μηχανικών Πληροφορικής και Ηλεκτρονικών Συστημάτων του ΔΙ.ΠΑ.Ε.

Η παρούσα εργασία αποτελεί πνευματική ιδιοκτησία του φοιτητή Θεόδωρος Διαμαντίδης που την εκπόνησε. Στο πλαίσιο της πολιτικής ανοικτής πρόσβασης, ο συγγραφέας/δημιουργός εκχωρεί στο Διεθνές Πανεπιστήμιο της Ελλάδος άδεια χρήσης του δικαιώματος αναπαραγωγής, διανομής, παρουσίασης στο κοινό και ψηφιακής διάχυσης της εργασίας διεθνώς, σε ηλεκτρονική μορφή και σε οποιοδήποτε μέσο, για διδακτικούς και ερευνητικούς σκοπούς, άνευ ανταλλάγματος. Η ανοικτή πρόσβαση στο πλήρες κείμενο της εργασίας, δεν σημαίνει καθ' οιονδήποτε τρόπο παραχώρηση δικαιωμάτων διανοητικής ιδιοκτησίας του συγγραφέα/δημιουργού, ούτε επιτρέπει την αναπαραγωγή, αναδημοσίευση, αντιγραφή, πώληση, εμπορική χρήση, διανομή, έκδοση, μεταφόρτωση (downloading), ανάρτηση (uploading), μετάφραση, τροποποίηση με οποιοδήποτε τρόπο, τμηματικά ή περιληπτικά της εργασίας, χωρίς τη ρητή προηγούμενη έγγραφη συναίνεση του συγγραφέα/δημιουργού.

Η έγκριση της πτυχιακής εργασίας από το Τμήμα Μηχανικών Πληροφορικής και Ηλεκτρονικών Συστημάτων του Διεθνούς Πανεπιστημίου της Ελλάδος, δεν υποδηλώνει απαραίτητα και αποδοχή των απόψεων του συγγραφέα, εκ μέρους του Τμήματος.

«Η πρόοδος είναι αποτέλεσμα συνεχούς πειραματισμού και βελτίωσης.»
- Συγγραφέας

Abstract

This thesis presents the AI Financial Dashboard, a web application designed for analyzing and visualizing financial trends for stocks and cryptocurrencies. The application allows users to ask questions such as “Is it a good time to buy?” or “What is the current trend?” and provides answers based on data from the past 30 days. Data is retrieved via the Finnhub and CoinGecko APIs, while artificial intelligence techniques are used to extract useful insights and trends. The platform offers interactive dashboards and visualizations, enabling users to explore market trends in a user-friendly and intuitive manner. This work contributes to the application of AI in understanding financial data and provides a practical tool for students, analysts, and investors seeking data-driven decision support.

Περίληψη

Η παρούσα πτυχιακή εργασία παρουσιάζει το AI Financial Dashboard, μια εφαρμογή ιστού σχεδιασμένη για ανάλυση και οπτικοποίηση τάσεων χρηματοοικονομικών δεδομένων για μετοχές και κρυπτονομίσματα. Η εφαρμογή επιτρέπει στους χρήστες να θέτουν ερωτήσεις τύπου «είναι καλή στιγμή να αγοράσω;» ή «ποια είναι η τρέχουσα τάση;», και παρέχει απαντήσεις βασισμένες σε δεδομένα των τελευταίων 30 ημερών. Τα δεδομένα αντλούνται μέσω των API Finnhub και CoinGecko, ενώ η ανάλυση αξιοποιεί τεχνικές τεχνητής νοημοσύνης για την εξαγωγή χρήσιμων πληροφοριών και συμπερασμάτων. Η πλατφόρμα προσφέρει διαδραστικά dashboards και γραφήματα, επιτρέποντας στους χρήστες να εξερευνούν τις τάσεις των αγορών με φιλικό και διαισθητικό τρόπο. Η εργασία αυτή συνεισφέρει στην αξιοποίηση των δυνατοτήτων AI στην κατανόηση των χρηματοοικονομικών δεδομένων, παρέχοντας ένα πρακτικό εργαλείο για φοιτητές, αναλυτές και επενδυτές που επιθυμούν υποστήριξη στη λήψη αποφάσεων βασισμένη σε δεδομένα.

Αφιέρωση

Αρχικά, θα ήθελα να εκφράσω την ευγνωμοσύνη μου στην οικογένεια και τους φίλους μου για την αμέριστη υποστήριξη και την συνεχή ενθάρρυνσή τους καθ' όλη τη διάρκεια των σπουδών μου. Οι πολύτιμες στιγμές που μοιραστήκαμε μου προσέφεραν δύναμη και χαρά σε κάθε βήμα αυτής της διαδρομής.

Επιπλέον, θα ήθελα να ευχαριστήσω θερμά τον επιβλέποντα της πτυχιακής μου εργασίας, Επίκουρο Καθηγητή κ. Στέφανο Ουγιάρογλου, ο οποίος με στήριξε από την πρώτη στιγμή, δείχνοντας εμπιστοσύνη τόσο ως προς το θέμα της εργασίας όσο και στην ολοκληρωμένη πορεία εκπόνησής της. Η καθοδήγησή του, οι συμβουλές του και η συνεχής υποστήριξη αποτέλεσαν καθοριστικούς παράγοντες για την επιτυχή ολοκλήρωση αυτής της εργασίας.

Περιεχόμενα

Περίληψη	iv
Αφιέρωση	v
Περιεχόμενα.....	vi
Γλωσσάρι	ix
1 Εισαγωγή.....	1
1.1 Χρηματοοικονομικές Αγορές	1
1.1.1 Μετοχές (Stocks)	2
1.1.2 Κρυπτονομίσματα (Cryptocurrencies)	3
1.1.3 Τάσεις Αγοράς και Ανάλυση Δεδομένων	4
1.1.4 Ομοιότητες και Διαφορές μεταξύ Stocks και Cryptocurrencies	5
1.2 Δεδομένα Χρηματοοικονομικών Αγορών	6
1.2.1 Πηγές και Τύποι Δεδομένων	6
1.2.2 Περίοδος Δεδομένων και Ιστορικά Στοιχεία	7
1.2.3 Όγκος Δεδομένων και Συχνότητα Ενημέρωσης	7
1.2.4 Η Αξία των Δεδομένων για Επενδυτές	8
1.2.5 Πηγές Δεδομένων της Εργασίας (Finnhub & CoinGecko)	8
1.3 Κίνητρο	9
1.4 Συνεισφορά	9
1.5 Οργάνωση Πτυχιακής	10
2 Εισαγωγή στις Βιβλιοθήκες	11
2.1 Βασικές Δυνατότητες της Βιβλιοθήκης CoinGecko	12
2.2 Βασικές Δυνατότητες της Βιβλιοθήκης Finnhub	12
2.2.1 Παραδείγματα Χρήσης του CoinGecko API	13
2.2.2 Ανάκτηση Ιστορικών Δεδομένων Τιμών	14
2.2.3 Παραδείγματα Χρήσης Finnhub API	15
2.3 API: Πλεονεκτήματα και Περιορισμοί	17
2.3.1 Finnhub API	17
2.3.2 CoinGecko API	17
3 Τεχνολογίες	18
3.1 TypeScript	18

3.1.1	Τι προσφέρει η TypeScript	18
3.1.2	Κενά της JavaScript που καλύπτει η TypeScript	19
3.1.3	Ρόλος της TypeScript στην παρούσα εφαρμογή	20
3.2	JSX	20
3.2.1	Βασικά Χαρακτηριστικά	20
3.2.2	Ο ρόλος του JSX στην εφαρμογή	21
3.3	React	21
3.3.1	Εισαγωγή	21
3.3.2	Βασικά Χαρακτηριστικά	21
3.4	Next.js App Router	23
3.4.1	Βασική Λειτουργία του App Router	23
3.4.2	Server και Client Components	23
3.4.3	Διαχείριση Δεδομένων	23
3.4.4	Ρόλος στην παρούσα εφαρμογή	24
3.5	Tailwind CSS	24
3.6	Backend	25
3.6.1	MongoDB Atlas	25
3.7	Διαχείριση Ασύγχρονων Ροών και Background Εργασιών	26
3.7.1	Inngest	26
4	Σχεδίαση και Υλοποίηση του AI Financial Dashboard	27
4.1	Λειτουργικές Απαιτήσεις	27
4.1.1	Προβολή Κεντρικής Σελίδας (Main Dashboard)	27
4.1.2	Παραμετροποίηση Διαγραμμάτων	27
4.1.3	Πρόσβαση χωρίς Πιστοποίηση Χρήστη	28
4.1.4	Λειτουργίες για Πιστοποιημένους Χρήστες	28
4.1.5	AI Προβλέψεις	29
4.2	Αρχιτεκτονική της Εφαρμογής	30
4.3	Υλοποίηση Backend	33
4.3.1	Βάση Δεδομένων και Backend Αποθήκευση	33
4.3.2	Συλλογή crypto	35
4.3.3	Συλλογή coin_charts	36
4.3.4	Συλλογή users	37
4.3.5	Συλλογή stock-news	38
4.3.6	Συλλογή ai_runs	39
4.4	Υλοποίηση Frontend	39
4.4.1	Η κεντρική δομή της εφαρμογής — το αρχείο App.tsx και layout.tsx	39
4.4.2	Components	42
4.4.3	Παράδειγμα React Component: CryptoChart	46
5	Παρουσίαση εφαρμογής AI Financial Dashbaord	53
5.1	Αρχική Σελίδα (Dashboard)	53
5.2	Διαχείριση Αγαπημένων Στοιχείων (Manage Favorites)	55
5.3	Λειτουργίες Σελίδας Manage Favorites	56
5.3.1	Πίνακας Αγαπημένων Στοιχείων (Watchlist Table)	56

5.3.2	Πίνακας Ειδοποιήσεων (Alerts Table)	57
5.3.3	Ενότητα Ανάλυσης με Τεχνητή Νοημοσύνη (AI Analysis Section)	58
5.3.4	Ημερήσιες Ενημερώσεις (Daily News)	59
6	Συμπεράσματα και Μελλοντική Εργασία	60
6.1	Συμπεράσματα	60
6.2	Μελλοντική Επέκταση	60

Γλωσσάρι

DeFi (Αποκεντρωμένα Χρηματοοικονομικά Προϊόντα) : Αποτελεί ένα οικοσύστημα χρηματοοικονομικών εφαρμογών που λειτουργούν σε blockchain, χωρίς κεντρικούς διαμεσολαβητές όπως τράπεζες ή χρηματιστηριακές εταιρείες. Παρέχει υπηρεσίες όπως δανεισμό, δανεισμό με εγγύηση, ανταλλαγές tokens και αποταμίευση, με πλήρη διαφάνεια και άμεση πρόσβαση σε όλους τους χρήστες.

Smart Contracts (Έξυπνα Συμβόλαια) : Πρόκειται για προγράμματα που εκτελούνται αυτόματα σε ένα blockchain όταν πληρούνται προκαθορισμένες συνθήκες. Χρησιμοποιούνται για αυτοματοποιημένες συναλλαγές, διαχείριση ψηφιακών περιουσιακών στοιχείων και εφαρμογές DeFi, εξαλείφοντας την ανάγκη για κεντρικό ενδιάμεσο.

Blockchain (Αλυσίδα Μπλοκ) : Είναι μια αποκεντρωμένη και αμετάβλητη βάση δεδομένων που αποθηκεύει συναλλαγές και πληροφορίες σε «μπλοκ», τα οποία συνδέονται με χρονολογική σειρά. Η τεχνολογία blockchain χρησιμοποιείται για την ασφάλεια, τη διαφάνεια και την ακεραιότητα σε κρυπτονομίσματα και εφαρμογές DeFi.

Sentiment (Συναίσθημα Αγοράς) : Αναφέρεται στη συλλογική διάθεση και ψυχολογία των επενδυτών σε μια αγορά, η οποία μπορεί να είναι θετική, αρνητική ή ουδέτερη. Το sentiment επηρεάζει την τιμή των μετοχών και των κρυπτονομισμάτων και χρησιμοποιείται συχνά σε ανάλυση δεδομένων και τεχνητή νοημοσύνη για πρόβλεψη τάσεων.

Κατάλογος σχημάτων

1.1	Τυπικό χρηματιστηριακό διάγραμμα μετοχής από TradingView, που δείχνει τις διακυμάνσεις της τιμής σε μια καθορισμένη χρονική περίοδο. Πηγή: TradingView.	2
1.2	Διάγραμμα τιμής κρυπτονομίσματος Bitcoin, που δείχνει τις μεταβολές της τιμής κατά τις τελευταίες 30 ημέρες. Πηγή: CoinGecko.	3
1.3	Είκονα αγοράς Bullish vs Bearish. Πηγή: Funexclub.	4
2.1	Chart απο την εφαρμογή, παρουσιάζει την μετάβολη στην τιμή ενός νομίσματος. Πηγή: AI Financial Dashboard.	15
3.1	Τι προσφέρει η typescript. Πηγή: laxminarayana31.	19
3.2	Διάγραμμα που δείχνει πώς ο Virtual DOM αντιπροσωπεύει και ενημερώνει μόνο τα στοιχεία που αλλάζουν στο πραγματικό DOM.	22
4.1	Flowchart της ιστοσελιδας. Πηγή: AI Financial Dashboard.	32
5.1	Αρχικη σελιδα τις ιστοσελιδας. Πηγή: AI Financial Dashboard.	54
5.2	Σελίδα login-register. Πηγή: AI Financial Dashboard.	55
5.3	Σελίδα watchlist. Πηγή: AI Financial Dashboard.	56

Κατάλογος πινάκων

4.1	Δομή εγγράφου της συλλογής <code>crypto</code>	35
4.2	Δομή εγγράφου της συλλογής <code>coin_charts</code>	36
4.3	Δομή εγγράφου της συλλογής <code>users</code>	37
4.4	Δομή εγγράφου της συλλογής <code>stock-news</code>	38
4.5	Δομή εγγράφου της συλλογής <code>ai_runs</code>	39

Κεφάλαιο 1

Εισαγωγή

1.1 Χρηματοοικονομικές Αγορές

Οι χρηματοοικονομικές αγορές αποτελούν θεμελιώδες κομμάτι της παγκόσμιας οικονομίας, καθώς παρέχουν ένα πλαίσιο για την ανταλλαγή χρηματοοικονομικών προϊόντων μεταξύ επενδυτών, εταιρειών και θεσμικών φορέων. Μέσω αυτών των αγορών, οι συμμετέχοντες μπορούν να επενδύσουν, να διαχειριστούν τον κίνδυνο και να αποκτήσουν κεφάλαια για ανάπτυξη. Οι αγορές αυτές περιλαμβάνουν τόσο παραδοσιακά χρηματοοικονομικά προϊόντα, όπως μετοχές, ομόλογα και παράγωγα, όσο και ψηφιακά μέσα και κρυπτονομίσματα, τα οποία έχουν εμφανιστεί τα τελευταία χρόνια ως ανεξάρτητες και αποκεντρωμένες αγορές.

Η λειτουργία των χρηματοοικονομικών αγορών στηρίζεται σε βασικές αρχές προσφοράς και ζήτησης, διαφάνειας και ρευστότητας. Η ανάλυση των δεδομένων αυτών των αγορών αποτελεί καθοριστικό εργαλείο για την κατανόηση των τάσεων και την υποστήριξη λήψης επενδυτικών αποφάσεων. Τεχνικές όπως η τεχνική ανάλυση, η θεμελιώδης ανάλυση και οι σύγχρονες μέθοδοι της τεχνητής νοημοσύνης επιτρέπουν στους επενδυτές να εξάγουν χρήσιμα συμπεράσματα από ιστορικά και τρέχοντα δεδομένα.

Η διείσδυση των ψηφιακών τεχνολογιών έχει επιφέρει σημαντικές αλλαγές στη μορφή και τη διαχείριση των αγορών. Τα κρυπτονομίσματα και οι πλατφόρμες trading υψηλής συχνότητας προσφέρουν νέες δυνατότητες για άμεση πρόσβαση σε δεδομένα, αυτοματοποιημένη ανάλυση και γρήγορες αποφάσεις, αλλά ταυτόχρονα δημιουργούν προκλήσεις όσον αφορά την ασφάλεια και τον έλεγχο των αγορών. Η σωστή αξιοποίηση των δεδομένων αυτών αποτελεί βασικό στόχο κάθε επενδυτή και αναλυτή στο σύγχρονο χρηματοοικονομικό περιβάλλον.

1.1.1 Μετοχές (Stocks)

Οι μετοχές αποτελούν χρηματοοικονομικούς τίτλους που αντιπροσωπεύουν ένα μέρος της ιδιοκτησίας μιας εταιρείας. Κατέχοντας μετοχές, οι επενδυτές αποκτούν δικαίωμα συμμετοχής στα κέρδη της εταιρείας, κυρίως μέσω μερισμάτων, αλλά και δυνατότητα κέρδους από την αύξηση της τιμής της μετοχής στην αγορά. Οι μετοχές διαπραγματεύονται κυρίως σε οργανωμένες αγορές, όπως τα χρηματιστήρια, όπου η τιμή τους καθορίζεται από την προσφορά και τη ζήτηση.



Σχήμα 1.1: Τυπικό χρηματιστηριακό διάγραμμα μετοχής από TradingView, που δείχνει τις διακυμάνσεις της τιμής σε μια καθορισμένη χρονική περίοδο. Πηγή: [TradingView](#).

Η αξία μιας μετοχής επηρεάζεται από πλήθος παραγόντων. Σημαντικοί παράγοντες είναι τα οικονομικά αποτελέσματα της εταιρείας, η γενικότερη κατάσταση της οικονομίας, οι τάσεις στον κλάδο στον οποίο δραστηριοποιείται η εταιρεία, και η ψυχολογία των επενδυτών. Οι αναλυτές χρησιμοποιούν διάφορες μεθόδους για την αξιολόγηση των μετοχών: η θεμελιώδης ανάλυση εξετάζει τα οικονομικά μεγέθη και τις προοπτικές της εταιρείας, ενώ η τεχνική ανάλυση βασίζεται σε ιστορικά δεδομένα τιμών και όγκου συναλλαγών για την πρόβλεψη μελλοντικών κινήσεων.

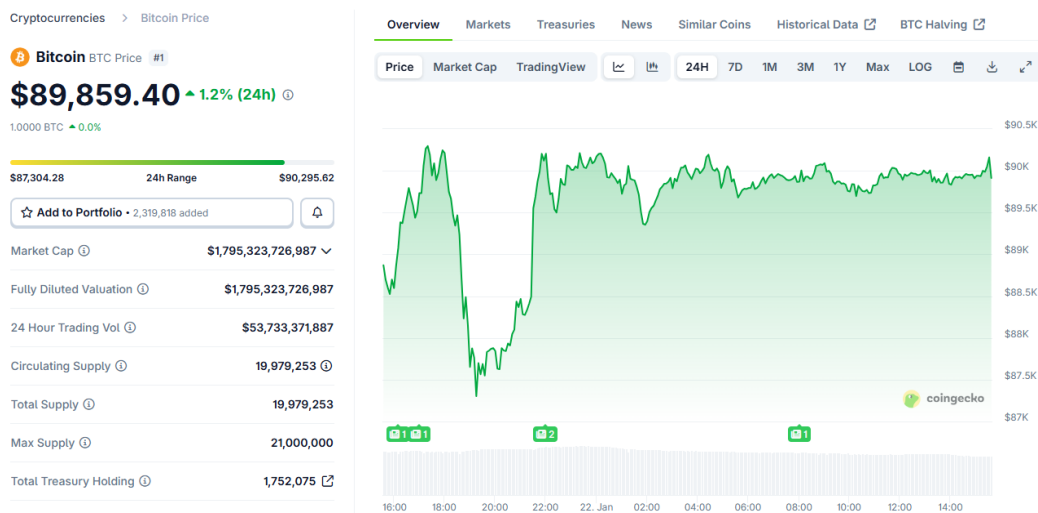
Οι μετοχές παρέχουν στους επενδυτές ευκαιρίες για κερδοφορία, αλλά συνοδεύονται και από κινδύνους, ειδικά σε περιόδους υψηλής αστάθειας της αγοράς. Η χρήση τεχνητής νοημοσύνης και αλγοριθμικών εργαλείων ανάλυσης επιτρέπει την εξαγωγή χρήσιμων συμπερασμάτων από μεγάλα σύνολα δεδομένων, βοηθώντας τους επενδυτές να λαμβάνουν τεκμηριωμένες αποφάσεις βασισμένες σε τάσεις και πρότυπα της αγοράς.

1.1.2 Κρυπτονομίσματα (Cryptocurrencies)

Τα κρυπτονομίσματα είναι ψηφιακά ή εικονικά νομίσματα που χρησιμοποιούν κρυπτογραφικές τεχνικές για την ασφαλή δημιουργία και διαχείριση συναλλαγών, ανεξάρτητα από κεντρική τράπεζα ή κυβερνητικό οργανισμό. Τα πιο γνωστά παραδείγματα είναι το Bitcoin και το Ethereum, τα οποία έχουν αναδειχθεί σε διεθνώς αποδεκτά μέσα συναλλαγών και επένδυσης.

Η διαπραγμάτευση των κρυπτονομισμάτων γίνεται σε εξειδικευμένες πλατφόρμες ανταλλαγής (crypto exchanges), όπως το Binance, το Coinbase και το Kraken, ενώ πολλές πλατφόρμες προσφέρουν και real-time charts για ανάλυση τιμών και τάσεων. Οι επενδυτές αξιοποιούν τεχνική ανάλυση, θεμελιώδη ανάλυση και πλέον αλγοριθμικές μεθόδους τεχνητής νοημοσύνης για να προβλέψουν πιθανές κινήσεις της αγοράς.

Τα κρυπτονομίσματα παρουσιάζουν υψηλή αστάθεια, γεγονός που σημαίνει ότι οι τιμές τους μπορεί να αυξομειωθούν σημαντικά μέσα σε σύντομο χρονικό διάστημα. Παράλληλα, προσφέρουν νέες δυνατότητες όπως αποκεντρωμένα χρηματοοικονομικά προϊόντα (DeFi)* και έξυπνα συμβόλαια (smart contracts)*, τα οποία ανοίγουν δρόμους για καινοτομία στον χρηματοοικονομικό τομέα.



Σχήμα 1.2: Διάγραμμα τιμής κρυπτονομίσματος Bitcoin, που δείχνει τις μεταβολές της τιμής κατά τις τελευταίες 30 ημέρες. Πηγή: [CoinGecko](#).

Η ανάλυση των δεδομένων αυτών των ψηφιακών αγορών παρέχει στους επενδυτές πολύτιμες πληροφορίες για τη λήψη αποφάσεων, ενώ η χρήση αλγορίθμων τεχνητής νοημοσύνης επιτρέπει την αναγνώριση μοτίβων, τάσεων και πιθανών κινδύνων σε πραγματικό χρόνο. Η εκμετάλλευση αυτών των εργαλείων είναι κρίσιμη για κάθε αναλυτή ή επενδυτή που επιθυμεί τεκμηριωμένη στρατηγική στον χώρο των κρυπτονομισμάτων.

1.1.3 Τάσεις Αγοράς και Ανάλυση Δεδομένων

Οι τάσεις της αγοράς αποτελούν σημαντικό εργαλείο για την κατανόηση της κατεύθυνσης των τιμών σε μετοχές και κρυπτονομίσματα. Οι βασικές κατηγορίες τάσεων είναι:

- **Bullish:** Ανοδική τάση με σταθερή αύξηση τιμών.
- **Bearish:** Πτωτική τάση με σταδιακή μείωση τιμών.
- **Sideways:** Σταθερή αγορά χωρίς σαφή κατεύθυνση.

Η ανάλυση των δεδομένων της αγοράς γίνεται μέσω ιστορικών δεδομένων τιμών, όγκου συναλλαγών και δεικτών μεταβλητότητας (volatility). Η τεχνική ανάλυση αξιοποιεί γραφήματα και δείκτες όπως RSI, MACD και κινητούς μέσους όρους (Moving Averages) για την πρόβλεψη πιθανών κινήσεων. Επιπλέον, η εφαρμογή τεχνητής νοημοσύνης επιτρέπει την αναγνώριση μοτίβων, την πρόβλεψη τάσεων και την υποστήριξη λήψης επενδυτικών αποφάσεων σε πραγματικό χρόνο.



Σχήμα 1.3: Είκονα αγοράς Bullish vs Bearish. Πήγη: [Funexclub](#).

1.1.4 Ομοιότητες και Διαφορές μεταξύ Stocks και Cryptocurrencies

Παρά τις βασικές διαφορές τους, οι αγορές μετοχών και κρυπτονομισμάτων παρουσιάζουν και ορισμένες ομοιότητες:

- Και οι δύο είναι χρηματοοικονομικά περιουσιακά στοιχεία που μπορούν να αγοραστούν, να πωληθούν ή να ανταλλαχθούν σε αγορές.
- Η τιμή τους επηρεάζεται από προσφορά και ζήτηση, καθώς και από ψυχολογικούς παράγοντες των επενδυτών.
- Και οι δύο αγορές χρησιμοποιούν δεδομένα για ανάλυση τάσεων, διαχείριση κινδύνου και υποστήριξη λήψης αποφάσεων.
- Η τεχνητή νοημοσύνη και οι αλγοριθμικές τεχνικές είναι χρήσιμες και στις δύο αγορές για πρόβλεψη τάσεων και μοτίβων.

Οι κυριότερες διαφορές περιλαμβάνουν:

- **Χρόνος Λειτουργίας:** Οι μετοχές συνήθως διαπραγματεύονται σε συγκεκριμένο ωράριο χρηματιστηρίου, ενώ τα κρυπτονομίσματα λειτουργούν 24/7.
- **Σταθερότητα και Ρίσκο:** Οι μετοχές είναι συνήθως πιο σταθερές και υπόκεινται σε ρυθμιστικά πλαίσια, ενώ τα κρυπτονομίσματα παρουσιάζουν υψηλή αστάθεια και περιορισμένη ρύθμιση.
- **Πληροφορίες για Ανάλυση:** Στις μετοχές υπάρχουν εκθέσεις κερδών, μερισμάτων και οικονομικές αναφορές. Στα crypto, η ανάλυση βασίζεται σε blockchain* δεδομένα, όγκο συναλλαγών και sentiment από κοινότητες.
- **Τεχνολογικά εργαλεία:** Τα κρυπτονομίσματα αξιοποιούν αποκεντρωμένα χρηματοοικονομικά προϊόντα (DeFi)* και έξυπνα συμβόλαια (smart contracts)*, που δεν υπάρχουν στις παραδοσιακές μετοχές.

Η κατανόηση των ομοιοτήτων και διαφορών αυτών των αγορών επιτρέπει στους επενδυτές και αναλυτές να εφαρμόζουν κατάλληλες στρατηγικές ανάλογα με το προφίλ ρίσκου, το χρονικό ορίζοντα και τα διαθέσιμα εργαλεία ανάλυσης.

1.2 Δεδομένα Χρηματοοικονομικών Αγορών

Η ανάλυση των χρηματοοικονομικών αγορών βασίζεται σε αξιόπιστα δεδομένα που καταγράφουν την πορεία των τιμών, τον όγκο των συναλλαγών και άλλους σημαντικούς δείκτες. Τα δεδομένα αυτά αποτελούν τη βάση για τη λήψη τεκμηριωμένων επενδυτικών αποφάσεων, την εκτίμηση κινδύνου και την πρόβλεψη τάσεων.

Στην παρούσα εργασία, η συλλογή και επεξεργασία δεδομένων αφορά τόσο τις παραδοσιακές αγορές μετοχών όσο και την αγορά κρυπτονομισμάτων. Η σωστή κατανόηση των δεδομένων αυτών είναι κρίσιμη για την εφαρμογή τεχνικών ανάλυσης, όπως η τεχνική ανάλυση, η στατιστική ανάλυση και οι μέθοδοι τεχνητής νοημοσύνης, οι οποίες χρησιμοποιούνται για την υποστήριξη των επενδυτών και την παροχή χρήσιμων πληροφοριών μέσα από την πλατφόρμα AI Financial Dashboard.

Στις επόμενες υποενότητες παρουσιάζονται αναλυτικά οι πηγές των δεδομένων, η μορφή τους, η περίοδος κάλυψης, ο όγκος και η συχνότητα ενημέρωσης, καθώς και η αξία τους για τη λήψη αποφάσεων.

1.2.1 Πηγές και Τύποι Δεδομένων

Τα δεδομένα που χρησιμοποιούνται στην εφαρμογή AI Financial Dashboard προέρχονται από επίσημες και αξιόπιστες πηγές, οι οποίες παρέχουν ενημερωμένα και ακριβή στοιχεία για χρηματοοικονομικές αγορές. Οι κύριες πηγές δεδομένων περιλαμβάνουν:

- **Finnhub API:** Παρέχει δεδομένα για μετοχές, δείκτες και συναλλαγματικές ισοτιμίες. Τα δεδομένα περιλαμβάνουν τιμές ανοίγματος, κλεισίματος, υψηλής και χαμηλής τιμής, όγκο συναλλαγών, καθώς και ιστορικά στοιχεία ανά ημέρα. Τα δεδομένα αυτά είναι απαραίτητα για την ανάλυση τάσεων, την αξιολόγηση κινδύνου και την εκτίμηση πιθανών επενδυτικών ευκαιριών.
- **CoinGecko API:** Παρέχει δεδομένα για κρυπτονομίσματα, όπως τιμές αγοράς και πώλησης, market cap, όγκο συναλλαγών και ιστορικά δεδομένα. Επιτρέπει την παρακολούθηση της δυναμικής των crypto αγορών και τη χρήση τους σε μοντέλα τεχνητής νοημοσύνης για πρόβλεψη τάσεων.

Οι τύποι δεδομένων που συλλέγονται περιλαμβάνουν:

- Τιμές ανοίγματος και κλεισίματος (Open / Close Prices)
- Υψηλές και χαμηλές τιμές (High / Low)
- Όγκο συναλλαγών (Volume)
- Market capitalization (μόνο για κρυπτονομίσματα)
- Ιστορικά δεδομένα για τις τελευταίες 30 ημέρες

Η επιλογή των συγκεκριμένων πηγών διασφαλίζει την αξιοπιστία και την επικαιρότητα των δεδομένων, ενώ οι τύποι δεδομένων που συλλέγονται παρέχουν πλήρες πλαίσιο για ανάλυση τάσεων και υποστήριξη λήψης αποφάσεων από επενδυτές και αναλυτές.

1.2.2 Περίοδος Δεδομένων και Ιστορικά Στοιχεία

Η περίοδος των δεδομένων αποτελεί κρίσιμο παράγοντα για την αξιοπιστία και την αποτελεσματικότητα της χρηματοοικονομικής ανάλυσης. Στην παρούσα εργασία, η συλλογή δεδομένων επικεντρώνεται σε ιστορικά στοιχεία έως και των τελευταίων 30 ημερών. Η συγκεκριμένη χρονική περίοδος επιλέχθηκε ώστε να αποτυπώνει τις πρόσφατες τάσεις της αγοράς και τη βραχυπρόθεσμη συμπεριφορά των χρηματοοικονομικών προϊόντων.

Τα ιστορικά δεδομένα περιλαμβάνουν ημερήσιες τιμές ανοίγματος, κλεισίματος, υψηλής και χαμηλής τιμής, καθώς και τον όγκο συναλλαγών. Για τα κρυπτονομίσματα, συμπεριλαμβάνονται επιπλέον δεδομένα όπως η κεφαλαιοποίηση της αγοράς (market capitalization), τα οποία συμβάλλουν στην πληρέστερη κατανόηση της δυναμικής της αγοράς.

Η χρήση δεδομένων βραχυπρόθεσμης περιόδου είναι ιδιαίτερα χρήσιμη σε εφαρμογές ανάλυσης σε πραγματικό ή σχεδόν πραγματικό χρόνο, όπως το AI Financial Dashboard. Τα δεδομένα αυτά επιτρέπουν την ταχύτερη ανίχνευση μεταβολών, την αναγνώριση μοτίβων και την υποστήριξη αλγορίθμων τεχνητής νοημοσύνης που στοχεύουν στην πρόβλεψη τάσεων και στη λήψη επενδυτικών αποφάσεων.

Παράλληλα, η χρήση ιστορικών δεδομένων διασφαλίζει ότι τα μοντέλα ανάλυσης βασίζονται σε πραγματικά παρατηρημένες συμπεριφορές της αγοράς, μειώνοντας τον κίνδυνο λανθασμένων εκτιμήσεων και αυξάνοντας την αξιοπιστία των αποτελεσμάτων.

1.2.3 Όγκος Δεδομένων και Συχνότητα Ενημέρωσης

Ο όγκος των δεδομένων και η συχνότητα ενημέρωσής τους αποτελούν καθοριστικούς παράγοντες για την ανάλυση και την αποτελεσματικότητα των συστημάτων υποστήριξης επενδυτικών αποφάσεων. Οι σύγχρονες χρηματοοικονομικές αγορές παράγουν τεράστιες ποσότητες δεδομένων σε συνεχή βάση, καθώς οι τιμές και οι όγκοι συναλλαγών μεταβάλλονται δυναμικά μέσα στην ημέρα.

Στην παρούσα εργασία, τα δεδομένα που χρησιμοποιούνται προέρχονται από αξιόπιστες διαδικτυακές πλατφόρμες και συλλέγονται μέσω αυτοματοποιημένων διεπαφών προγραμματισμού εφαρμογών (APIs). Η ενημέρωση των δεδομένων πραγματοποιείται σε ημερήσια βάση.

Ο μεγάλος όγκος δεδομένων καθιστά απαραίτητη τη χρήση διαδικασιών φιλτραρίσματος πριν την αποθήκευσή τους στη βάση δεδομένων, ώστε να διατηρούνται μόνο τα απαραίτητα και χρήσιμα στοιχεία για την ανάλυση. Με τον τρόπο αυτό επιτυγχάνεται πιο αποδοτική διαχείριση των δεδομένων και μειώνεται ο αποθηκευτικός φόρτος.

Η σωστή ισορροπία μεταξύ όγκου δεδομένων και συχνότητας ενημέρωσης είναι κρίσιμη για την ανάπτυξη εφαρμογών ανάλυσης σε πραγματικό χρόνο, όπως το AI Financial Dashboard. Μέσω αυτής της προσέγγισης, διασφαλίζεται ότι οι αλγόριθμοι τεχνητής νοημοσύνης τροφοδοτούνται με επαρκή και επικαιροποιημένα δεδομένα, ενισχύοντας την ακρίβεια και την αξιοπιστία των παραγόμενων αποτελεσμάτων.

1.2.4 Η Αξία των Δεδομένων για Επενδυτές

Τα χρηματοοικονομικά δεδομένα αποτελούν βασικό εργαλείο για τη λήψη τεκμηριωμένων επενδυτικών αποφάσεων. Μέσα από την ανάλυση ιστορικών και τρεχουσών δεδομένων, οι επενδυτές μπορούν να εντοπίζουν τάσεις, να αξιολογούν τον κίνδυνο και να εκτιμούν πιθανές μελλοντικές κινήσεις της αγοράς. Η αξιοποίηση φιλτραρισμένων και αξιόπιστων δεδομένων συμβάλλει στη μείωση της αβεβαιότητας και στην υποστήριξη πιο αποτελεσματικών στρατηγικών επένδυσης.

1.2.5 Πηγές Δεδομένων της Εργασίας (Finnhub & CoinGecko)

Η πρόσβαση στα χρηματοοικονομικά δεδομένα για την παρούσα πτυχιακή πραγματοποιείται μέσω των API Finnhub και CoinGecko. Τα συγκεκριμένα API παρέχουν δεδομένα για μετοχές, κρυπτονομίσματα, τιμές ανοίγματος/κλεισίματος, όγκο συναλλαγών και ιστορικά στοιχεία.

Η αναλυτική παρουσίαση των API και του τρόπου ενσωμάτωσής τους στην εφαρμογή γίνεται στο Κεφάλαιο 2, όπου παρουσιάζονται οι βασικές λειτουργίες τους και η διαχείριση των δεδομένων για την τροφοδότηση του AI Financial Dashboard.

1.3 Κίνητρο

Στα προηγούμενα υποκεφάλαια παρουσιάστηκαν οι βασικές έννοιες σχετικά με τις χρηματοοικονομικές αγορές, τις μετοχές και τα κρυπτονομίσματα, καθώς και η σημασία της ανάλυσης των δεδομένων τους για λήψη τεκμηριωμένων επενδυτικών αποφάσεων. Είναι εμφανές ότι οι αγορές αυτές παράγουν τεράστιο όγκο δεδομένων καθημερινά, ο οποίος μπορεί να αποκαλύψει τάσεις και ευκαιρίες για επενδυτές και αναλυτές.

Ωστόσο, η ευχρηστία στην παρακολούθηση των δεδομένων αποτελεί σημαντική πρόκληση. Η παρουσίαση πολλών πληροφοριών ταυτόχρονα, χωρίς κατάλληλη οργάνωση ή φιλτράρισμα, μπορεί να είναι συντριπτική (overwhelming) για τον χρήστη, δυσχεραίνοντας την ταχεία κατανόηση των τάσεων της αγοράς και τη λήψη αποφάσεων. Επιπλέον, η δυνατότητα παρακολούθησης σημαντικών μεταβολών σε πραγματικό χρόνο, όπως η ενημέρωση για τιμές που φτάνουν καθορισμένα όρια (thresholds), αποτελεί ανάγκη για επενδυτές και αναλυτές που δεν μπορούν να ελέγχουν συνεχώς τα δεδομένα.

Αυτά τα κενά δημιούργησαν το κίνητρο για την ανάπτυξη του AI Financial Dashboard: μια εύχρηστη και διαδραστική πλατφόρμα που παρέχει πρόσβαση σε δεδομένα μετοχών και κρυπτονομισμάτων, προσφέρει οπτικοποίηση τάσεων και υποστηρίζει την κατανόηση της αγοράς ακόμη και από μη ειδικούς χρήστες.

1.4 Συνεισφορά

Η παρούσα εργασία συνεισφέρει στην αξιοποίηση και κατανόηση των χρηματοοικονομικών δεδομένων μέσω του AI Financial Dashboard, μιας ολοκληρωμένης και εύχρηστης πλατφόρμας ανάλυσης και οπτικοποίησης. Η εφαρμογή προσφέρει πρόσβαση σε δεδομένα μετοχών και κρυπτονομισμάτων, συνδυάζοντας διαδραστικά dashboards και γραφήματα που επιτρέπουν την απλή και κατανοητή παρουσίαση σύνθετων πληροφοριών.

Ένα από τα κύρια χαρακτηριστικά της εφαρμογής είναι η δυνατότητα άμεσης ενημέρωσης των χρηστών μέσω email όταν οι τιμές μετοχών ή κρυπτονομισμάτων φτάσουν σε προκαθορισμένα όρια (thresholds), επιτρέποντας την παρακολούθηση σημαντικών μεταβολών σε πραγματικό χρόνο χωρίς την ανάγκη συνεχούς ελέγχου των δεδομένων. Παράλληλα, η παρουσίαση πολλών δεδομένων σε οργανωμένο και φιλικό περιβάλλον μειώνει την υπερφόρτωση πληροφοριών και διευκολύνει την κατανόηση των τάσεων της αγοράς.

Επιπλέον, η εφαρμογή ενσωματώνει τεχνικές τεχνητής νοημοσύνης για την ανάλυση των δεδομένων, παρέχοντας στους χρήστες τη δυνατότητα εξαγωγής χρήσιμων συμπερασμάτων και τάσεων από μεγάλους όγκους δεδομένων. Με αυτόν τον τρόπο, το AI Financial Dashboard προσφέρει πρακτικό εργαλείο για φοιτητές, αναλυτές και επενδυτές, επιτρέποντάς τους να λαμβάνουν τεκμηριωμένες αποφάσεις βασισμένες σε δεδομένα.

Τέλος, η εργασία καλύπτει κενά που παρατηρούνται σε υπάρχουσες λύσεις, συνδυάζοντας ευ-

χρηστιά, διαδραστικότητα και προηγμένες τεχνολογίες ανάλυσης, παρέχοντας μια ολοκληρωμένη πλατφόρμα για την παρακολούθηση και κατανόηση των χρηματοοικονομικών αγορών, ακόμη και από χρήστες χωρίς προηγούμενη εμπειρία στον προγραμματισμό ή στην ανάλυση δεδομένων.

1.5 Οργάνωση Πτυχιακής

Η παρούσα πτυχιακή εργασία έχει οργανωθεί σε επιμέρους κεφάλαια με σκοπό να καθοδηγήσει τον αναγνώστη από τα βασικά στοιχεία της εφαρμογής μέχρι τις λεπτομέρειες της υλοποίησης και την παρουσίαση των αποτελεσμάτων.

- **Κεφάλαιο 1 – Εισαγωγή:** Παρουσιάζεται το αντικείμενο της εργασίας, οι στόχοι και το περιβάλλον στο οποίο αναπτύχθηκε η εφαρμογή.
- **Κεφάλαιο 2 – Εισαγωγή στις βιβλιοθήκες και τεχνολογίες:** Περιγράφονται οι τεχνολογίες που χρησιμοποιήθηκαν στο frontend και backend, όπως TypeScript, React, Next.js, MongoDB και Inngest.
- **Κεφάλαιο 3 – Τεχνολογίες:** Αναλύονται οι κύριες τεχνολογίες που χρησιμοποιούνται στην εφαρμογή, με αναφορά στις δυνατότητες και τη συνεισφορά τους στο project.
- **Κεφάλαιο 4 – Σχεδίαση και Υλοποίηση:** Παρουσιάζονται οι λειτουργικές απαιτήσεις, η αρχιτεκτονική της εφαρμογής, η υλοποίηση του frontend και του backend, καθώς και η διαχείριση δεδομένων.
- **Κεφάλαιο 5 – Παρουσίαση της εφαρμογής:** Εξηγούνται οι δυνατότητες που προσφέρει η εφαρμογή στον χρήστη, περιγράφοντας τη διαδραστική εμπειρία, τα dashboards, τα alerts και τα καθημερινά νέα.
- **Κεφάλαιο 6 – Συμπεράσματα και Προοπτικές:** Γίνονται συνοπτικά σχόλια για τα αποτελέσματα της εργασίας, η αξιολόγηση της εφαρμογής και οι μελλοντικές βελτιώσεις που μπορούν να γίνουν.
- **Βιβλιογραφία:** Περιλαμβάνει όλες τις πηγές που χρησιμοποιήθηκαν για την ανάπτυξη της εφαρμογής και τη συγγραφή της εργασίας.

Κεφάλαιο 2

Εισαγωγή στις Βιβλιοθήκες

Στο κεφάλαιο αυτό παρουσιάζονται οι βασικές βιβλιοθήκες και τα APIs που χρησιμοποιήθηκαν για την άντληση χρηματοοικονομικών δεδομένων στο πλαίσιο της παρούσας εργασίας. Η αξιόπιστη και οργανωμένη πρόσβαση σε δεδομένα αγοράς αποτελεί κρίσιμο παράγοντα για την ανάπτυξη εφαρμογών ανάλυσης, καθώς η ποιότητα των δεδομένων επηρεάζει άμεσα την εγκυρότητα των συμπερασμάτων που προκύπτουν.

Για τις ανάγκες του AI Financial Dashboard επιλέχθηκαν δύο βασικές πηγές δεδομένων: το Finnhub API για δεδομένα μετοχών και παραδοσιακών χρηματοοικονομικών αγορών, και το CoinGecko API για δεδομένα κρυπτονομισμάτων. Οι δύο αυτές πλατφόρμες παρέχουν σύγχρονα, αξιόπιστα και εύκολα προσβάσιμα δεδομένα μέσω διαδικτυακών διεπαφών (REST APIs), επιτρέποντας την αυτοματοποιημένη συλλογή πληροφοριών από την εφαρμογή.

Η χρήση APIs επιτρέπει τη δυναμική ανάκτηση δεδομένων, όπως τιμές, όγκους συναλλαγών και ιστορικά στοιχεία. Παράλληλα, διευκολύνει τη συνεχή ενημέρωση της εφαρμογής με τα πιο πρόσφατα δεδομένα της αγοράς, διατηρώντας το σύστημα ευέλικτο και επεκτάσιμο.

Στα επόμενα υποκεφάλαια παρουσιάζονται αναλυτικά τα χαρακτηριστικά κάθε βιβλιοθήκης, οι τύποι δεδομένων που παρέχουν και ο τρόπος με τον οποίο ενσωματώθηκαν στην εφαρμογή.

2.1 Βασικές Δυνατότητες της Βιβλιοθήκης CoinGecko

Το CoinGecko API αποτελεί μία από τις πιο δημοφιλείς και αξιόπιστες πηγές δεδομένων για την αγορά κρυπτονομισμάτων. Παρέχει δωρεάν πρόσβαση σε μεγάλο όγκο πληροφοριών σχετικά με τιμές, όγκους συναλλαγών, κεφαλαιοποίηση αγοράς και ιστορικά δεδομένα για χιλιάδες ψηφιακά νομίσματα.

Μία από τις βασικές δυνατότητες της βιβλιοθήκης είναι η παροχή δεδομένων τιμών σε πραγματικό χρόνο, επιτρέποντας την άμεση παρακολούθηση της αγοράς. Επιπλέον, το API προσφέρει ιστορικά δεδομένα τιμών, τα οποία είναι απαραίτητα για την ανάλυση τάσεων και τη μελέτη της συμπεριφοράς των αγορών σε συγκεκριμένα χρονικά διαστήματα. Στο πλαίσιο της παρούσας εργασίας, αξιοποιούνται δεδομένα των τελευταίων 30 ημερών για την εξαγωγή συμπερασμάτων σχετικά με τη βραχυπρόθεσμη τάση ενός κρυπτονομίσματος.

Το CoinGecko παρέχει επίσης πληροφορίες όπως:

- Κεφαλαιοποίηση αγοράς (market capitalization)
- Όγκο συναλλαγών (trading volume)
- Ποσοστιαίες μεταβολές τιμής σε διαφορετικά χρονικά διαστήματα
- Κατάταξη κρυπτονομισμάτων με βάση τη δημοτικότητα ή την αξία τους

Ένα σημαντικό πλεονέκτημα του CoinGecko API είναι ότι δεν απαιτεί σύνθετη διαδικασία αυθεντικοποίησης για βασική χρήση, γεγονός που διευκολύνει την ενσωμάτωσή του σε εκπαιδευτικές και ερευνητικές εφαρμογές. Παράλληλα, η δομημένη μορφή των δεδομένων (JSON) επιτρέπει την εύκολη επεξεργασία τους μέσω προγραμματιστικών εργαλείων.

Στην εφαρμογή AI Financial Dashboard, το CoinGecko χρησιμοποιείται για την ανάκτηση δεδομένων τιμών και τάσεων κρυπτονομισμάτων, τα οποία τροφοδοτούν τα διαδραστικά γραφήματα, τις αναλύσεις και τις απαντήσεις του συστήματος σε ερωτήματα χρηστών σχετικά με την πορεία της αγοράς.

2.2 Βασικές Δυνατότητες της Βιβλιοθήκης Finnhub

Το Finnhub API αποτελεί μια σύγχρονη πλατφόρμα παροχής χρηματοοικονομικών δεδομένων για παραδοσιακές αγορές, όπως μετοχές, δείκτες και συνάλλαγμα. Χρησιμοποιείται ευρέως σε εφαρμογές ανάλυσης αγορών, καθώς προσφέρει πρόσβαση σε δεδομένα τιμών, ιστορικά στοιχεία και πληροφορίες που σχετίζονται με τη συμπεριφορά των χρηματοοικονομικών μέσων.

Μία από τις βασικές δυνατότητες του Finnhub είναι η παροχή δεδομένων τιμών μετοχών σχεδόν σε πραγματικό χρόνο, επιτρέποντας την παρακολούθηση των μεταβολών της αγοράς. Παράλληλα, το API παρέχει ιστορικά δεδομένα τιμών (candlestick data), τα οποία είναι απαραίτητα για την ανάλυση τάσεων και τη μελέτη της πορείας μιας μετοχής σε συγκεκριμένο χρονικό διάστημα. Στην παρούσα εργασία, αξιοποιούνται δεδομένα των τελευταίων 30 ημερών για την ανάλυση της βραχυπρόθεσμης τάσης.

Το Finnhub προσφέρει επιπλέον πληροφορίες όπως:

- Τιμές ανοίγματος, κλεισίματος, μέγιστης και ελάχιστης τιμής
- Όγκο συναλλαγών
- Ιστορικά δεδομένα για διαφορετικά χρονικά διαστήματα
- Βασικές πληροφορίες για εταιρείες και σύμβολα μετοχών

Ένα σημαντικό χαρακτηριστικό του Finnhub είναι η παροχή δεδομένων σε δομημένη μορφή (JSON), γεγονός που διευκολύνει την επεξεργασία και ενσωμάτωσή τους σε εφαρμογές ανάλυσης. Η χρήση API keys επιτρέπει ελεγχόμενη και ασφαλή πρόσβαση στα δεδομένα.

Στο AI Financial Dashboard, το Finnhub χρησιμοποιείται για την ανάκτηση δεδομένων μετοχών, τα οποία αξιοποιούνται για τη δημιουργία γραφημάτων, την ανάλυση τάσεων και την υποστήριξη των απαντήσεων που παρέχει η εφαρμογή σε ερωτήματα χρηστών σχετικά με την πορεία μιας μετοχής.

2.2.1 Παραδείγματα Χρήσης του CoinGecko API

Για τη χρήση του CoinGecko API απαιτείται αρχικά η δημιουργία ενός **API key** μέσω της πλατφόρμας του CoinGecko. Το κλειδί αυτό χρησιμοποιείται για την ταυτοποίηση της εφαρμογής και αποστέλλεται σε κάθε αίτημα (*request*) προς το API μέσω των HTTP headers. Με αυτόν τον τρόπο επιτυγχάνεται ελεγχόμενη πρόσβαση στα δεδομένα.

Σε κάθε αίτημα προς το API, το κλειδί συμπεριλαμβάνεται στο header, επιτρέποντας την ανάκτηση δεδομένων σχετικά με κρυπτονομίσματα, τιμές αγοράς, όγκο συναλλαγών και άλλα στατιστικά στοιχεία.

Παρακάτω παρουσιάζεται παράδειγμα κλήσης του API για την ανάκτηση δεδομένων αγοράς κρυπτονομισμάτων:

```
const res = await fetch(
  "https://api.coingecko.com/api/v3/coins/markets?vs_currency=usd&order=market_cap_desc",
  {
    headers: {
      "x-cg-demo-api-key": apiKey,
      Accept: "application/json",
    },
  },
);

const data = await res.json();

const filteredData: Coin[] = data.map((coin: any) => ({
  id: coin.id,
  symbol: coin.symbol,
```

```

    name: coin.name,
    image: coin.image,
    current_price: coin.current_price,
    change: coin.price_change_percentage_24h,
    market_cap: coin.market_cap,
    last_updated: coin.last_updated,
  }));

```

Στο παραπάνω παράδειγμα:

- Η παράμετρος **vs_currency=usd** ορίζει ότι οι τιμές επιστρέφονται σε δολάρια.
- Το **order=market_cap_desc** ταξινομεί τα κρυπτονομίσματα με βάση τη χρηματιστηριακή αξία (market cap).
- Το **per_page=100** καθορίζει τον αριθμό αποτελεσμάτων ανά σελίδα.
- Το **page=1** δηλώνει την πρώτη σελίδα αποτελεσμάτων.
- Το header **x-cg-demo-api-key** περιέχει το API key της εφαρμογής.

Η απάντηση του API επιστρέφεται σε μορφή **JSON** και περιλαμβάνει μεγάλο πλήθος πεδίων για κάθε κρυπτονομίσμα. Ωστόσο, για τις ανάγκες της εφαρμογής δεν χρησιμοποιούνται όλα τα δεδομένα. Πραγματοποιείται διαδικασία φιλτραρίσματος και μετασχηματισμού (mapping), κατά την οποία επιλέγονται μόνο τα απαραίτητα πεδία, ώστε να μειώνεται ο όγκος πληροφορίας και να διευκολύνεται η επεξεργασία και αποθήκευση.

Συγκεκριμένα, από κάθε αντικείμενο κρυπτονομίσματος διατηρούνται στοιχεία όπως το αναγνωριστικό, το σύμβολο, το όνομα, η εικόνα, η τρέχουσα τιμή, η ημερήσια ποσοστιαία μεταβολή, η κεφαλαιοποίηση αγοράς και η τελευταία ενημέρωση. Η διαδικασία αυτή υλοποιείται με μετατροπή των δεδομένων σε μια νέα δομή αντικειμένων που χρησιμοποιείται εσωτερικά στην εφαρμογή.

2.2.2 Ανάκτηση Ιστορικών Δεδομένων Τιμών

Εκτός από τα τρέχοντα στοιχεία της αγοράς, η βιβλιοθήκη CoinGecko API επιτρέπει και την ανάκτηση ιστορικών δεδομένων τιμών ενός κρυπτονομίσματος. Αυτό είναι ιδιαίτερα σημαντικό για την ανάλυση τάσεων, τη δημιουργία γραφημάτων και την εφαρμογή τεχνικών προβλεψής.

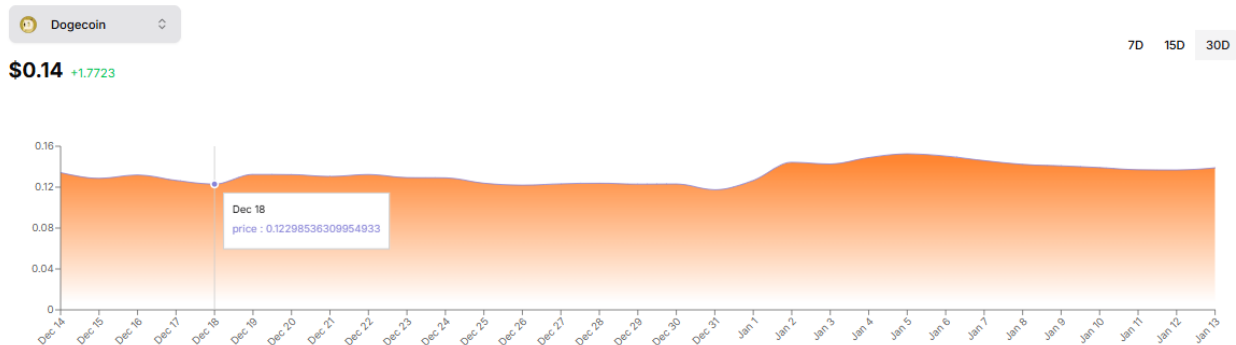
Στο παρακάτω παράδειγμα πραγματοποιείται αίτημα προς το endpoint `market_chart`, το οποίο επιστρέφει δεδομένα τιμής για συγκεκριμένο χρονικό διάστημα. Στην προκειμένη περίπτωση ζητούνται δεδομένα των τελευταίων 30 ημερών για ένα κρυπτονομίσμα, χρησιμοποιώντας ως νόμισμα αναφοράς το δολάριο ΗΠΑ (USD).

```
const chart = await fetch(
  `https://api.coingecko.com/api/v3/coins/dogecoin/
  market_chart?vs_currency=usd&days=30`,
  {
    headers: {
      "x-cg-demo-api-key": apiKey,
      Accept: "application/json",
    },
  },
)
).then((res) => res.json())
```

Η απάντηση του API περιλαμβάνει χρονοσειρές δεδομένων (time series), όπως:

- ιστορικές τιμές (prices),
- κεφαλαιοποίηση αγοράς (market_caps),
- όγκο συναλλαγών (total_volumes).

Τα δεδομένα αυτά χρησιμοποιούνται για τη δημιουργία γραφημάτων εξέλιξης της τιμής, επιτρέποντας την οπτική κατανόηση των διακυμάνσεων της αγοράς και την υποστήριξη διαδικασιών τεχνικής ανάλυσης.



Σχήμα 2.1: Chart απο την εφαρμογή, παρουσιάζει την μετάβολη στην τιμή ενός νομίσματος. Πηγή: [AI Financial Dashboard](#).

2.2.3 Παραδείγματα Χρήσης Finnhub API

Για τη χρήση του Finnhub API απαιτείται η δημιουργία προσωπικού API key μέσω της επίσημης πλατφόρμας. Το κλειδί αυτό αποστέλλεται σε κάθε HTTP αίτημα και επιτρέπει την ασφαλή πρόσβαση στα δεδομένα.

Ανάκτηση Χρηματοοικονομικών Ειδήσεων Εταιρείας

Ένα σενάριο χρήσης είναι η ανάκτηση ειδήσεων που σχετίζονται με μια συγκεκριμένη μετοχή. Οι ειδήσεις αυτές μπορούν να αξιοποιηθούν για ανάλυση συναισθήματος (sentiment analysis) και κατανόηση της επίδρασης της ειδησεογραφίας στην τιμή της μετοχής.

Παράδειγμα αιτήματος για ειδήσεις που αφορούν τη μετοχή της Apple (AAPL):

```
https://finnhub.io/api/v1/company-news?symbol=AAPL&from=2025-12-01
&to=2025-12-31&token=YOUR_API_KEY
```

Η απάντηση του API επιστρέφει λίστα άρθρων ειδήσεων σε μορφή JSON, όπου κάθε αντικείμενο περιλαμβάνει πληροφορίες όπως τίτλο άρθρου, πηγή, ημερομηνία δημοσίευσης, περίληψη και σύνδεσμο προς το πλήρες κείμενο. Τα δεδομένα αυτά μπορούν να φιλτραριστούν και να αξιοποιηθούν για την εξαγωγή δεικτών συναισθήματος ή για την ενημέρωση του χρήστη σχετικά με σημαντικές εξελίξεις που επηρεάζουν τη συγκεκριμένη μετοχή.

```
{
  "_id": "xxxxxxxxxxxxxxxxxxxx",
  "category": "company",
  "datetime": 1767200604,
  "headline": "Plexo Capital's Lo Toney: AI
'inference economics' important to watch ...",
  "id": 137958449,
  "image": "https://image.cnbcfm.com/...",
  "related": "AAPL",
  "source": "CNBC",
  "summary": "Plexo Capital's Lo Toney joins
'Closing Bell Overtime' to talk ...",
  "url": "https://finnhub.io/api/news?id=..."
}
```

Κάθε πεδίο της απόκρισης έχει συγκεκριμένο ρόλο:

- **_id**: Μοναδικό αναγνωριστικό εγγραφής στη βάση δεδομένων.
- **category**: Κατηγορία είδησης (π.χ. company, market).
- **datetime**: Χρονική σήμανση δημοσίευσης (Unix timestamp).
- **headline**: Τίτλος του άρθρου.
- **source**: Πηγή της είδησης (π.χ. CNBC, Yahoo).
- **summary**: Σύντομη περίληψη του άρθρου.
- **related**: Το σύμβολο της μετοχής με την οποία σχετίζεται η είδηση.
- **image**: Σύνδεσμος προς εικόνα του άρθρου.
- **url**: Σύνδεσμος προς το πλήρες άρθρο.

2.3 API: Πλεονεκτήματα και Περιορισμοί

2.3.1 Finnhub API

Ένα από τα σημαντικότερα πλεονεκτήματά της είναι η δυνατότητα πραγματικού χρόνου ενημέρωσης, κάτι που καθιστά την Finnhub ιδανική για εφαρμογές που απαιτούν άμεση παρακολούθηση των αγορών. Επιπλέον, η καλή τεκμηρίωση και οι έτοιμες βιβλιοθήκες για διάφορες γλώσσες προγραμματισμού διευκολύνουν την ενσωμάτωσή της σε projects.

Ωστόσο, η πλήρης αξιοποίηση της Finnhub περιορίζεται από την ανάγκη συνδρομής για προχωρημένα δεδομένα και τον περιορισμένο αριθμό κλήσεων στο δωρεάν πλάνο. Επιπλέον, η εξάρτηση από τους εξωτερικούς servers του Finnhub σημαίνει ότι η αξιοπιστία της εφαρμογής συνδέεται άμεσα με τη διαθεσιμότητα της υπηρεσίας.

2.3.2 CoinGecko API

Ένα από τα βασικά πλεονεκτήματά του είναι ότι τα βασικά δεδομένα είναι διαθέσιμα χωρίς συνδρομή, διευκολύνοντας την άμεση ενσωμάτωσή του σε εφαρμογές ή αναλύσεις. Επιπλέον, υποστηρίζει τόσο real-time όσο και ιστορικά δεδομένα, καθιστώντας το χρήσιμο για μακροπρόθεσμες μελέτες αγοράς.

Ωστόσο, υπάρχουν περιορισμοί στη χρήση του CoinGecko, όπως ο περιορισμένος αριθμός κλήσεων ανά λεπτό για τον δωρεάν λογαριασμό και η έλλειψη δεδομένων σε επίπεδο order book, κάτι που μπορεί να περιορίσει την ανάλυση σε βάθος. Η αξιοπιστία των δεδομένων εξαρτάται επίσης από τη διαθεσιμότητα των servers του CoinGecko.

Κεφάλαιο 3

Τεχνολογίες

3.1 TypeScript

Η **TypeScript** είναι ένα υπερσύνολο (superset) της γλώσσας προγραμματισμού **JavaScript**, το οποίο προσθέτει μηχανισμούς στατικής τυποποίησης (static typing) και προηγμένα εργαλεία ανάπτυξης λογισμικού. Αναπτύχθηκε από τη Microsoft και μεταγλωττίζεται σε καθαρή JavaScript, επιτρέποντας την εκτέλεση του παραγόμενου κώδικα σε οποιοδήποτε περιβάλλον που υποστηρίζει JavaScript, όπως οι φυλλομετρητές (web browsers) και οι διακομιστές.

Σκοπός της TypeScript δεν είναι η αντικατάσταση της JavaScript, αλλά η επέκτασή της με λειτουργίες που διευκολύνουν την ανάπτυξη μεγάλων και σύνθετων εφαρμογών ιστού.

3.1.1 Τι προσφέρει η TypeScript

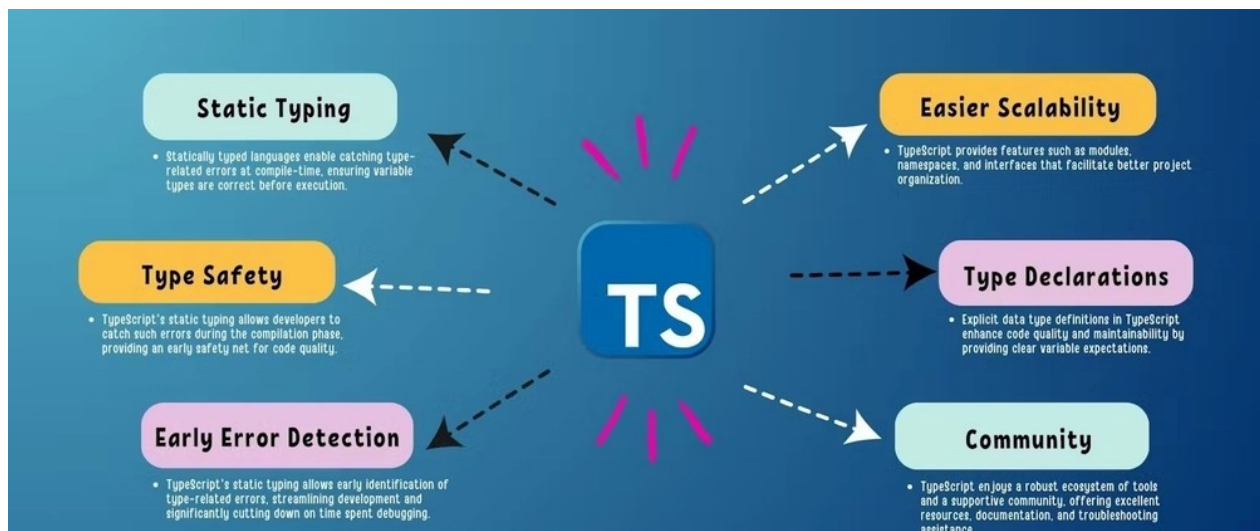
Η TypeScript εισάγει χαρακτηριστικά που απουσιάζουν από την κλασική JavaScript:

- **Στατική τυποποίηση (Static Typing):** Οι μεταβλητές, συναρτήσεις και αντικείμενα μπορούν να έχουν προκαθορισμένους τύπους δεδομένων (π.χ. `string`, `number`, `boolean`). Τα σφάλματα τύπων εντοπίζονται κατά τη μεταγλώττιση και όχι κατά την εκτέλεση.
- **Δομές δεδομένων μέσω Interfaces και Types:** Επιτρέπουν τον ορισμό δομημένων αντικειμένων και συμβολαίων (contracts) για τα δεδομένα μιας εφαρμογής.
- **Βελτιωμένη υποστήριξη αντικειμενοστραφούς προγραμματισμού:** Παρέχει σαφή υλοποίηση κλάσεων, κληρονομικότητας και modifiers πρόσβασης όπως `public`, `private` και `protected`.
- **Ανάπτυξη με εργαλεία (Tooling):** Προσφέρει καλύτερη αυτόματη συμπλήρωση (autocompletion), έλεγχο λαθών σε πραγματικό χρόνο και δυνατότητες αναδόμησης κώδικα (refactoring).

3.1.2 Κενά της JavaScript που καλύπτει η TypeScript

Παρόλο που η JavaScript είναι ιδιαίτερα ευέλικτη, παρουσιάζει αδυναμίες σε εφαρμογές μεγάλης κλίμακας. Η TypeScript έρχεται να καλύψει τα εξής κενά:

- **Δυναμική τυποποίηση:** Στη JavaScript τα σφάλματα τύπων εμφανίζονται κατά την εκτέλεση. Η TypeScript εισάγει στατικό έλεγχο τύπων πριν την εκτέλεση του προγράμματος.
- **Δυσκολία διαχείρισης μεγάλων κωδίκων:** Μέσω των interfaces και των modules, η TypeScript επιτρέπει πιο δομημένη αρχιτεκτονική.
- **Ασαφείς δομές δεδομένων:** Η JavaScript δεν επιβάλλει συγκεκριμένη δομή σε αντικείμενα. Η TypeScript επιτρέπει ρητό ορισμό τύπων και σχημάτων δεδομένων.
- **Περιορισμένη υποστήριξη από τα IDE:** Η στατική τυποποίηση προσφέρει καλύτερη συνεργασία με εργαλεία ανάπτυξης, βελτιώνοντας την παραγωγικότητα.



Σχήμα 3.1: Τι προσφέρει η typescript. Πηγή: [laxminarayana31](#).

3.1.3 Ρόλος της TypeScript στην παρούσα εφαρμογή

Στην εφαρμογή **AI Financial Dashboard**, η TypeScript:

- διασφαλίζει τη σωστή δομή των οικονομικών δεδομένων,
- μειώνει σφάλματα κατά την αλληλεπίδραση με εξωτερικά APIs,
- καθιστά πιο αξιόπιστη τη διαχείριση καταστάσεων (state management),
- βελτιώνει τη συντήρηση και επεκτασιμότητα του κώδικα.

Επομένως, λειτουργεί ως μηχανισμός ποιοτικού ελέγχου του λογισμικού, ιδιαίτερα σημαντικός σε μεγάλες εφαρμογές.

3.2 JSX

Το **JSX** (JavaScript XML) είναι μια σύνταξη που επιτρέπει στους προγραμματιστές να γράφουν HTML-like κώδικα μέσα σε αρχεία JavaScript. Είναι η τυπική μέθοδος για την περιγραφή της διεπαφής χρήστη (UI) σε εφαρμογές React και διευκολύνει τη σύνδεση του UI με την λογική της εφαρμογής.

3.2.1 Βασικά Χαρακτηριστικά

- **Σύνταξη παρόμοια με HTML:** Ο κώδικας JSX μοιάζει με HTML, αλλά μετατρέπεται σε JavaScript μέσω του *Babel* κατά το compile-time.
- **Ενσωμάτωση JavaScript:** Με τη χρήση αγκυλών {}, μπορούμε να ενσωματώσουμε δυναμικές εκφράσεις JavaScript μέσα στο JSX. Για παράδειγμα:

```
const name = "Θεόδωρος";
const greeting = <h1>Γειά σου, {name}!</h1>;
```

- **Components ως στοιχεία:** Τα JSX elements μπορούν να αντιπροσωπεύουν είτε HTML tags είτε React components, επιτρέποντας την επαναχρησιμοποίηση του κώδικα.
- **Διευκόλυνση Render Logic:** Το JSX επιτρέπει conditional rendering και mapping arrays σε elements, παρέχοντας ευέλικτη και declarative σύνταξη για UI:

```
const numbers = [1, 2, 3];
const listItems = numbers.map(n => <li key={n}>{n}</li>);
```

3.2.2 Ο ρόλος του JSX στην εφαρμογή

Στο *AI Financial Dashboard*, το JSX:

- επιτρέπει τη σύνδεση των δεδομένων με τη διεπαφή του χρήστη,
- καθιστά εύκολη την δημιουργία δυναμικών components για charts, tables και filters,
- διευκολύνει την ανάγνωση και συντήρηση του κώδικα, καθώς η HTML-like σύνταξη είναι πιο κατανοητή από καθαρό JavaScript DOM manipulation.

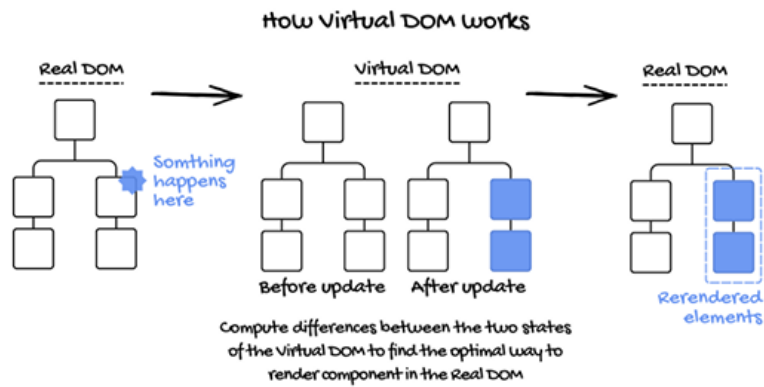
3.3 React

3.3.1 Εισαγωγή

Η **React** είναι μια βιβλιοθήκη JavaScript ανοιχτού κώδικα, η οποία χρησιμοποιείται για τη δημιουργία δυναμικών διεπαφών χρήστη (User Interfaces) σε εφαρμογές ιστού. Ανάπτυξη της React έγινε από τη Meta (πρώην Facebook) και έχει γίνει ένα από τα πιο δημοφιλή εργαλεία για την ανάπτυξη μοντέρνων web εφαρμογών.

3.3.2 Βασικά Χαρακτηριστικά

- **Component-Based Architecture:** Οι διεπαφές χωρίζονται σε ανεξάρτητα, επαναχρησιμοποιήσιμα *components*, καθιστώντας την ανάπτυξη πιο οργανωμένη και συντηρήσιμη.
- **Virtual DOM:** Η React χρησιμοποιεί ένα εικονικό DOM για να παρακολουθεί τις αλλαγές στο UI και να ενημερώνει αποδοτικά μόνο τα στοιχεία που έχουν αλλάξει, μειώνοντας το κόστος ανανέωσης της σελίδας.
- **Declarative Syntax:** Οι developers περιγράφουν πώς πρέπει να φαίνεται το UI σε κάθε κατάσταση της εφαρμογής, αντί να καθορίζουν βήμα-βήμα τις αλλαγές στο DOM.
- **State Management:** Η React επιτρέπει την αποθήκευση και παρακολούθηση της κατάστασης (state) ενός component, η οποία επηρεάζει τη συμπεριφορά και την εμφάνιση της διεπαφής.
- **Hooks:** Οι React Hooks (π.χ. `useState`, `useEffect`) επιτρέπουν στους developers να διαχειρίζονται state και side-effects σε λειτουργικά components χωρίς τη χρήση κλάσεων.



Σχήμα 3.2: Διάγραμμα που δείχνει πώς ο Virtual DOM αντιπροσωπεύει και ενημερώνει μόνο τα στοιχεία που αλλάζουν στο πραγματικό DOM.

3.4 Next.js App Router

Το **Next.js** είναι ένα σύγχρονο framework βασισμένο στη βιβλιοθήκη React, το οποίο παρέχει δυνατότητες ανάπτυξης εφαρμογών ιστού με υποστήριξη server-side rendering, στατικής δημιουργίας σελίδων και βελτιστοποιημένης δρομολόγησης (routing). Από την έκδοση 13 και έπειτα, εισήχθη ο μηχανισμός **App Router**, ο οποίος αντικαθιστά σταδιακά το παραδοσιακό σύστημα δρομολόγησης μέσω του φακέλου `pages`.

Το App Router αποτελεί τη σύγχρονη αρχιτεκτονική δρομολόγησης του Next.js και βασίζεται στη δομή φακέλων μέσα στον κατάλογο `app/`.

3.4.1 Βασική Λειτουργία του App Router

Στο App Router, κάθε φάκελος μέσα στον κατάλογο `app` αντιστοιχεί σε μία διαδρομή (route) της εφαρμογής. Τα αρχεία που χρησιμοποιούνται είναι:

- `page.tsx`: Ορίζει το περιεχόμενο της σελίδας για μία συγκεκριμένη διαδρομή.
- `layout.tsx`: Καθορίζει κοινή δομή (layout) για πολλές σελίδες, όπως κεφαλίδες, υποσέλιδα και πλοήγηση.
- `loading.tsx`: Εμφανίζεται κατά τη φόρτωση δεδομένων.
- `error.tsx`: Χειρίζεται σφάλματα σε επίπεδο διαδρομής.

Με αυτόν τον τρόπο, η δομή του συστήματος αρχείων καθορίζει αυτόματα τη δρομολόγηση της εφαρμογής.

3.4.2 Server και Client Components

Μία από τις σημαντικότερες καινοτομίες του App Router είναι ο διαχωρισμός μεταξύ:

- **Server Components**: Εκτελούνται στον διακομιστή, μειώνοντας τον όγκο JavaScript που αποστέλλεται στον φυλλομετρητή και βελτιώνοντας την απόδοση.
- **Client Components**: Εκτελούνται στον φυλλομετρητή και χρησιμοποιούνται όταν απαιτείται αλληλεπίδραση με τον χρήστη (π.χ. state, events).

Ο διαχωρισμός αυτός επιτρέπει καλύτερη αξιοποίηση των πόρων και βελτιωμένη εμπειρία χρήστη.

3.4.3 Διαχείριση Δεδομένων

Το App Router επιτρέπει την ανάκτηση δεδομένων απευθείας σε server components μέσω ασύγχρονων συναρτήσεων. Έτσι:

- μειώνεται η ανάγκη για ξεχωριστά API calls από τον browser,
- βελτιώνεται η απόδοση φόρτωσης,
- ενισχύεται η ασφάλεια, καθώς ευαίσθητα δεδομένα παραμένουν στον διακομιστή.

3.4.4 Ρόλος στην παρούσα εφαρμογή

Στην εφαρμογή *AI Financial Dashboard*, το Next.js App Router:

- οργανώνει τη δομή των σελίδων του dashboard,
- διαχειρίζεται layouts και κοινά στοιχεία διεπαφής,
- επιτρέπει server-side επεξεργασία δεδομένων,
- συμβάλλει στη βελτιστοποίηση της απόδοσης της εφαρμογής.

Η χρήση του App Router καθιστά την αρχιτεκτονική της εφαρμογής πιο κλιμακούμενη, αποδοτική και ευκολότερη στη συντήρηση.

3.5 Tailwind CSS

Το **Tailwind CSS** είναι ένα utility-first CSS framework που επιτρέπει τη δημιουργία διεπαφών χρήστη μέσω έτοιμων κλάσεων (utility classes) αντί της συγγραφής παραδοσιακών CSS αρχείων. Αντί να ορίζονται ξεχωριστά στυλ σε εξωτερικά αρχεία, τα στυλιστικά χαρακτηριστικά εφαρμόζονται απευθείας στα στοιχεία της HTML/JSX δομής.

Η φιλοσοφία του Tailwind βασίζεται στη σύνθεση μικρών, επαναχρησιμοποιήσιμων κλάσεων που αντιστοιχούν σε ιδιότητες CSS, όπως χρώματα, αποστάσεις, διαστάσεις και responsive συμπεριφορές. Αυτό οδηγεί σε ταχύτερη ανάπτυξη διεπαφών και συνεπή σχεδιασμό.

Βασικά Χαρακτηριστικά

- **Utility-First Προσέγγιση:** Στυλισμός μέσω μικρών, λειτουργικών κλάσεων.
- **Responsive Design:** Ενσωματωμένα breakpoints για διαφορετικές συσκευές.
- **Consistency:** Ενιαίο σύστημα αποστάσεων, χρωμάτων και τυπογραφίας.
- **Customization:** Παραμετροποίηση μέσω αρχείου `tailwind.config.js`.
- **Performance:** Αφαίρεση μη χρησιμοποιούμενων στυλ στην παραγωγή (purge).

3.6 Backend

3.6.1 MongoDB Atlas

Το **MongoDB Atlas** είναι μια πλήρως διαχειριζόμενη βάση δεδομένων NoSQL που βασίζεται σε έγγραφα (document-based) και προσφέρεται ως υπηρεσία cloud. Αποθηκεύει δεδομένα σε μορφή JSON-like εγγράφων (BSON) και υποστηρίζει δυναμικό σχήμα (schema-less), γεγονός που την καθιστά ιδανική για εφαρμογές που απαιτούν γρήγορη ανάπτυξη και ευέλικτη δομή δεδομένων.

Βασικά Χαρακτηριστικά

- **NoSQL Database:** Δεδομένα σε έγγραφα BSON αντί για πίνακες.
- **Cloud-Based Managed Service:** Αυτόματο scaling, backups και updates.
- **High Availability:** Replica sets για υψηλή διαθεσιμότητα.
- **Global Distribution:** Τοποθέτηση δεδομένων σε διαφορετικά κέντρα.
- **Flexible Queries:** Indexing, aggregation pipelines και transactions.

Σύνδεση με την εφαρμογή (MongoClient) Η σύνδεση της εφαρμογής με τη MongoDB Atlas γίνεται μέσω του επίσημου mongodb driver για Node.js.

```
// Εγκατάσταση: npm install mongodb
const { MongoClient } = require('mongodb');

// Connection string από MongoDB Atlas
const uri = "mongodb+srv://<username>:<password>@cluster0.mongodb.net/myData";

const client = new MongoClient(uri);

async function run() {
  try {
    await client.connect(); // Σύνδεση στη βάση
    const database = client.db('myDatabase');
    const collection = database.collection('financial_data');

    // Παράδειγμα ερώτησης: εύρεση πρώτου εγγράφου
    const doc = await collection.findOne({});
    console.log(doc);
  } finally {
    await client.close(); // Κλείσιμο σύνδεσης
  }
}

run().catch(console.dir);
```

Ρόλος της MongoDB Atlas στην εφαρμογή Το MongoDB Atlas χρησιμοποιείται για:

- Αποθήκευση οικονομικών δεδομένων και ιστορικών τιμών.
- Διαχείριση χρηστών και προτιμήσεων dashboard.
- Υποστήριξη API endpoints του backend για γρήγορη ανάκτηση και ανάλυση δεδομένων.

3.7 Διαχείριση Ασύγχρονων Ροών και Background Εργασιών

3.7.1 Inngest

Το **Inngest** είναι μια πλατφόρμα διαχείρισης event-driven ροών εργασίας (workflow orchestration) για εφαρμογές Node.js και serverless περιβάλλοντα. Επιτρέπει τον ορισμό συναρτήσεων που εκτελούνται ως απόκριση σε γεγονότα (events), υποστηρίζοντας μηχανισμούς όπως retries, καθυστερήσεις (delays), βήματα εκτέλεσης (steps) και διαχείριση σφαλμάτων.

Σε σύγχρονες web εφαρμογές, πολλές διεργασίες δεν πρέπει να εκτελούνται άμεσα στο request-response κύκλο (π.χ. ανάλυση δεδομένων, επεξεργασία οικονομικών δεικτών, ειδοποιήσεις, AI διεργασίες). Το Inngest επιτρέπει τη μεταφορά αυτών των διεργασιών σε background ροές, βελτιώνοντας:

- **Απόδοση (Performance):** Το frontend δεν περιμένει βαριές διεργασίες.
- **Αξιοπιστία (Reliability):** Αυτόματες επαναπροσπάθειες σε περίπτωση αποτυχίας.
- **Κλιμάκωση (Scalability):** Διαχείριση πολλαπλών ασύγχρονων εργασιών.
- **Παρατηρησιμότητα (Observability):** Παρακολούθηση της ροής εκτέλεσης.

Κεφάλαιο 4

Σχεδίαση και Υλοποίηση του AI Financial Dashboard

4.1 Λειτουργικές Απαιτήσεις

Οι λειτουργικές απαιτήσεις περιγράφουν τις βασικές δυνατότητες που προσφέρει το σύστημα στον τελικό χρήστη, καθώς και τη συμπεριφορά της εφαρμογής σε διάφορα σενάρια χρήσης. Συγκεκριμένα: Για την εφαρμογή «AI Financial Dashboard», οι απαιτήσεις περιλαμβάνουν

4.1.1 Προβολή Κεντρικής Σελίδας (Main Dashboard)

Η αρχική σελίδα της εφαρμογής παρέχει μια συνοπτική αλλά ολοκληρωμένη εικόνα της χρηματοοικονομικής κατάστασης της αγοράς.

- Προβάλλονται δύο βασικά διαγράμματα:
 - Ένα διάγραμμα σχετικό με την αγορά κρυπτονομισμάτων.
 - Ένα διάγραμμα που απεικονίζει χρηματοοικονομικούς δείκτες και μετοχές από κατηγορίες όπως *indices*, *financial*, *technology* και *services*.
- Παρουσιάζονται τα πιο πρόσφατα νέα της αγοράς, προσφέροντας ενημέρωση για σημαντικά γεγονότα που επηρεάζουν τις οικονομικές τάσεις.

Η σελίδα αυτή λειτουργεί ως κεντρικός πίνακας ελέγχου (dashboard), παρέχοντας γρήγορη επισκόπηση της αγοράς.

4.1.2 Παραμετροποίηση Διαγραμμάτων

Τα διαγράμματα υποστηρίζουν δυναμική παραμετροποίηση:

- Ο χρήστης μπορεί να επιλέξει χρονικό εύρος προβολής δεδομένων (π.χ. 1, 7 ή 30 ημέρες).
- Υποστηρίζεται επιλογή συγκεκριμένου χρηματοοικονομικού στοιχείου (μετοχή ή κρυπτονομίσμα) για προβολή ιστορικών δεδομένων.

Η λειτουργία αυτή επιτρέπει συγκριτική ανάλυση βραχυπρόθεσμων και μεσοπρόθεσμων τάσεων.

4.1.3 Πρόσβαση χωρίς Πιστοποίηση Χρήστη

Οι βασικές λειτουργίες προβολής δεδομένων και διαγραμμάτων είναι διαθέσιμες και σε μη πιστοποιημένους χρήστες. Έτσι, το σύστημα επιτρέπει ελεύθερη πρόσβαση σε πληροφορίες αγοράς χωρίς απαίτηση σύνδεσης (login).

4.1.4 Λειτουργίες για Πιστοποιημένους Χρήστες

Οι χρήστες που έχουν πραγματοποιήσει είσοδο στο σύστημα αποκτούν πρόσβαση σε πρόσθετες προσωποποιημένες λειτουργίες.

Διαχείριση Αγαπημένων (Favorites)

- Οι χρήστες μπορούν να προσθέτουν χρηματοοικονομικά στοιχεία (μετοχές ή κρυπτονομίσματα) στη λίστα αγαπημένων.
- Η λίστα αυτή χρησιμοποιείται για γρήγορη πρόσβαση σε συχνά παρακολουθούμενα στοιχεία.

Ημερήσια Εξατομικευμένη Ενημέρωση μέσω Email Το σύστημα υλοποιεί μηχανισμό αυτοματοποιημένης ημερήσιας ενημέρωσης, ο οποίος βασίζεται στη λίστα αγαπημένων χρηματοοικονομικών στοιχείων κάθε χρήστη. Σε προκαθορισμένο χρονικό σημείο ενεργοποιείται προγραμματισμένη διεργασία στο backend, η οποία αναλαμβάνει τη συλλογή και επεξεργασία πληροφοριών.

Αρχικά, ανακτώνται από τη βάση δεδομένων τα αγαπημένα στοιχεία (μετοχές και κρυπτονομίσματα) του χρήστη. Στη συνέχεια, πραγματοποιείται άντληση (fetch) πρόσφατων ειδησεογραφικών δεδομένων που σχετίζονται με τα συγκεκριμένα στοιχεία. Τα δεδομένα αυτά δεν αποστέλλονται αυτούσια· πριν την τελική παρουσίαση, διέρχονται από στάδιο επεξεργασίας μέσω μοντέλου τεχνητής νοημοσύνης **Gemini**. Το μοντέλο χρησιμοποιείται για:

- Σύνοψη (summarization) των ειδήσεων.
- Φιλτράρισμα σημαντικών πληροφοριών.
- Παραγωγή περιεκτικού και κατανοητού κειμένου για τον χρήστη.

Με τον τρόπο αυτό, το σύστημα μετατρέπει μεγάλο όγκο χρηματοοικονομικής πληροφορίας σε συνοπτική, δομημένη και προσωποποιημένη ενημέρωση.

Σε περίπτωση που ο χρήστης δεν έχει ορίσει αγαπημένα στοιχεία, ενεργοποιείται μηχανισμός εναλλακτικής λειτουργίας (fallback), όπου αποστέλλεται γενική σύνοψη των σημαντικότερων οικονομικών ειδήσεων της ημέρας. Έτσι διασφαλίζεται ότι κάθε χρήστης λαμβάνει χρήσιμη πληροφόρηση, ανεξάρτητα από το επίπεδο παραμετροποίησης του προφίλ του.

Τέλος, δημιουργείται εξατομικευμένο μήνυμα ηλεκτρονικού ταχυδρομείου, το οποίο αποστέλλεται αυτόματα στον χρήστη. Το email περιλαμβάνει τις συνοψισμένες ειδήσεις, βασικές μεταβολές τιμών και γενική εικόνα της αγοράς, προσφέροντας συνεχή ενημέρωση χωρίς να απαιτείται ενεργή είσοδος στην εφαρμογή.

Σελίδα Διαχείρισης Αγαπημένων Στην ενότητα προφίλ, ο χρήστης διαθέτει ειδική σελίδα διαχείρισης, όπου:

- Προβάλλονται αναλυτικές πληροφορίες για όλα τα αγαπημένα στοιχεία.
- Υπάρχει δυνατότητα προσθήκης ή αφαίρεσης στοιχείων από τη λίστα.

Σύστημα Ειδοποιήσεων (Alerts) Οι χρήστες μπορούν να δημιουργούν ειδοποιήσεις βασισμένες σε συνθήκες τιμής:

- Ορισμός συνθηκών, π.χ. *αν τιμή < 50*.
- Όταν η συνθήκη ικανοποιηθεί, ενεργοποιείται συμβάν (event).
- Το σύστημα αποστέλλει αναλυτικό email ειδοποίησης στον χρήστη.

Η λειτουργία αυτή βασίζεται σε μηχανισμό event-driven εκτέλεσης στο backend.

Επιπλέον, το σύστημα υποστηρίζει πλήρη διαχείριση των ειδοποιήσεων από τον χρήστη. Μέσω της σελίδας προφίλ, ο χρήστης μπορεί οποιαδήποτε στιγμή:

- Να τροποποιήσει (edit) τις συνθήκες μιας υπάρχουσας ειδοποίησης.
- Να διαγράψει (delete) ειδοποιήσεις που δεν επιθυμεί πλέον.

4.1.5 AI Προβλέψεις

Για κάθε στοιχείο της λίστας αγαπημένων, ο χρήστης μπορεί να εκτελέσει προκαθορισμένα ερωτήματα ανάλυσης. Τα ερωτήματα αυτά επεξεργάζονται από μοντέλο τεχνητής νοημοσύνης (Gemini AI), το οποίο:

- Αναλύει τα διαθέσιμα οικονομικά δεδομένα.
- Παράγει προβλέψεις ή ερμηνευτικές αναλύσεις.
- Επιστρέφει αποτελέσματα σε μορφή κατανοητή για τον χρήστη.

Οι προβλέψεις και οι αναλύσεις που παράγονται από το μοντέλο τεχνητής νοημοσύνης (Gemini) βασίζονται σε υπολογιστικά μοντέλα και επεξεργασία ιστορικών δεδομένων. Τα αποτελέσματα αυτά έχουν υποστηρικτικό χαρακτήρα και δεν αποτελούν επενδυτική συμβουλή ούτε εγγυώνται μελλοντική απόδοση χρηματοοικονομικών στοιχείων.

Οι χρηματοοικονομικές αγορές επηρεάζονται από παράγοντες που δεν είναι πάντα δυνατό να μοντελοποιηθούν πλήρως, όπως γεωπολιτικά γεγονότα, ψυχολογία επενδυτών και απρόβλεπτες οικονομικές εξελίξεις. Συνεπώς, οι πληροφορίες που παρέχονται από το σύστημα θα πρέπει να αξιοποιούνται ως εργαλείο ανάλυσης και όχι ως αποκλειστική βάση λήψης επενδυτικών αποφάσεων.

Η προσθήκη του μηχανισμού αυτού στο σύστημα έχει ερευνητικό και υποστηρικτικό χαρακτήρα, με στόχο τη διερεύνηση της εφαρμογής τεχνικών τεχνητής νοημοσύνης σε χρηματοοικονομικά δεδομένα.

4.2 Αρχιτεκτονική της Εφαρμογής

Όπως αναφέρθηκε και προηγουμένως, ο στόχος της εφαρμογής *AI Financial Dashboard* είναι να παρέχει στους χρήστες μια ολοκληρωμένη πλατφόρμα παρακολούθησης, ανάλυσης και κατανόησης χρηματοοικονομικών δεδομένων, συνδυάζοντας διαδραστική οπτικοποίηση, προσωποποιημένες λειτουργίες και τεχνικές τεχνητής νοημοσύνης.

Κατά την είσοδο στην αρχική σελίδα (Main Dashboard), ο χρήστης αποκτά άμεση εικόνα της κατάστασης των αγορών. Η σελίδα αυτή λειτουργεί ως κεντρικός κόμβος πληροφόρησης και περιλαμβάνει:

- Διαδραστικό γράφημα κρυπτονομισμάτων.
- Διαδραστικό γράφημα μετοχών και δεικτών (financial, technology, services κ.ά.).
- Συνοπτική προβολή των τελευταίων οικονομικών ειδήσεων.

Ο χρήστης μπορεί να παραμετροποιήσει τα γραφήματα επιλέγοντας χρονικό εύρος (1, 7 ή 30 ημέρες), καθώς και συγκεκριμένο χρηματοοικονομικό στοιχείο (μετοχή ή κρυπτονόμισμα). Οι βασικές αυτές λειτουργίες είναι διαθέσιμες και σε μη εγγεγραμμένους χρήστες.

Για τους εγγεγραμμένους χρήστες παρέχονται επιπλέον προσωποποιημένες δυνατότητες. Συγκεκριμένα, μπορούν να αποθηκεύουν χρηματοοικονομικά στοιχεία ως *αγαπημένα* (*favorites*), ώστε να έχουν άμεση πρόσβαση σε αυτά, καθώς και να λαμβάνουν καθημερινά ενημερωτικά emails με ειδήσεις σχετικές με τα επιλεγμένα στοιχεία. Οι ειδήσεις αυτές, πριν αποσταλούν, υποβάλλονται σε επεξεργασία από το μοντέλο τεχνητής νοημοσύνης Gemini, το οποίο δημιουργεί συνοπτικές περιλήψεις, βελτιώνοντας τη χρηστικότητα της πληροφορίας.

Η εφαρμογή υποστηρίζει επίσης σύστημα ειδοποιήσεων (alerts), όπου οι χρήστες μπορούν να ορίζουν συνθήκες τιμής (π.χ. *αν τιμή < 50*). Όταν μια συνθήκη ικανοποιηθεί, ενεργοποιείται συμβάν στο backend, το οποίο οδηγεί σε αποστολή αναλυτικού email ειδοποίησης. Οι χρήστες έχουν τη δυνατότητα δημιουργίας, επεξεργασίας και διαγραφής ειδοποιήσεων ανά πάσα στιγμή.

Επιπλέον, παρέχεται λειτουργία ερωτημάτων πρόβλεψης, όπου οι χρήστες μπορούν να επιλέγουν προκαθορισμένα ερωτήματα σχετικά με την πορεία ενός στοιχείου. Τα δεδομένα αποστέλλονται στο μοντέλο Gemini, το οποίο παράγει προβλεπτικές αναλύσεις και επεξηγηματικά

κείμενα.

Αρχιτεκτονική Συστήματος

Η αρχιτεκτονική της εφαρμογής ακολουθεί σύγχρονο web μοντέλο πολλαπλών επιπέδων και αποτελείται από τέσσερα βασικά τμήματα:

Frontend Το frontend έχει υλοποιηθεί με **Next.js (App Router)**, **React**, **TypeScript** και **Tailwind CSS**. Είναι υπεύθυνο για:

- Την εμφάνιση των διαδραστικών γραφημάτων.
- Τη διαχείριση της πλοήγησης και της εμπειρίας χρήστη.
- Την επικοινωνία με τα backend API routes.

Backend / API Layer Το backend υλοποιείται μέσω των API routes του Next.js και λειτουργεί ως ενδιάμεσο επίπεδο ανάμεσα στο frontend και τις υπηρεσίες δεδομένων. Διαχειρίζεται:

- Αιτήματα για χρηματοοικονομικά δεδομένα.
- Διαχείριση χρηστών και αγαπημένων στοιχείων.
- Δημιουργία και έλεγχο ειδοποιήσεων.
- Επικοινωνία με το AI μοντέλο Gemini.

Βάση Δεδομένων Για την αποθήκευση δεδομένων χρησιμοποιείται η **MongoDB Atlas**, μια cloud NoSQL βάση δεδομένων. Σε αυτή αποθηκεύονται:

- Πληροφορίες χρηστών.
- Λίστες αγαπημένων στοιχείων.
- Ρυθμίσεις ειδοποιήσεων.

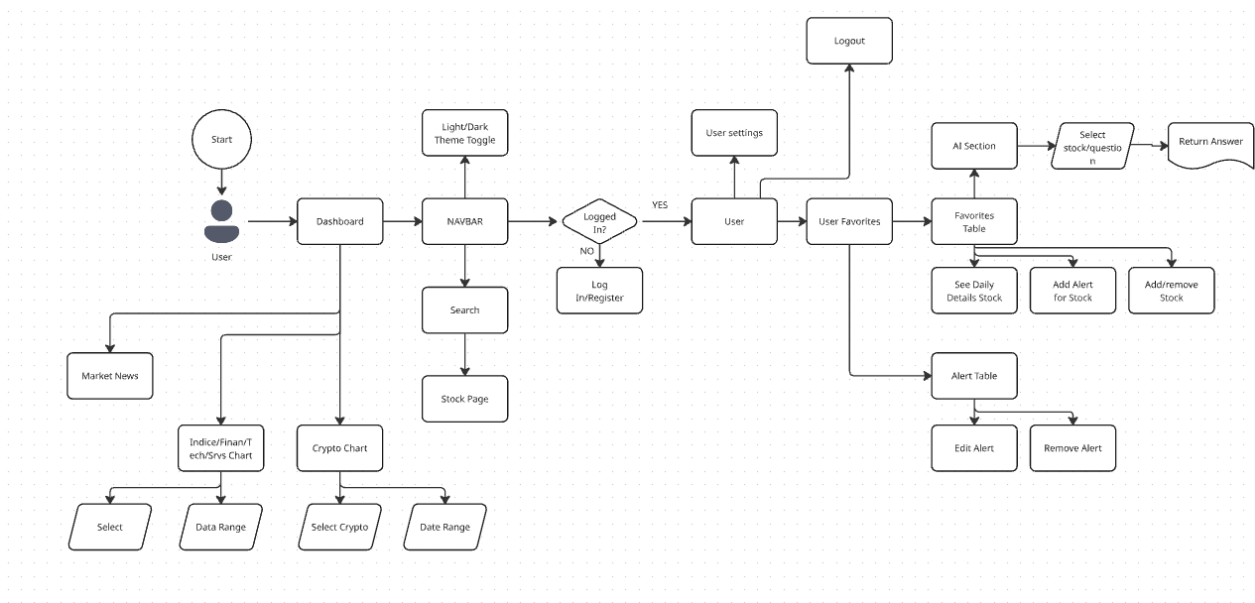
Event and AI Processing Layer Η εφαρμογή χρησιμοποιεί μηχανισμό event-driven εκτέλεσης μέσω της πλατφόρμας **Inngest**. Η τεχνολογία αυτή αναλαμβάνει:

- Την αποστολή email καλωσορίσματος.
- Την αποστολή καθημερινών email με ειδήσεις.
- Την ενεργοποίηση ειδοποιήσεων όταν ικανοποιούνται συνθήκες.
- Την εκτέλεση ροών εργασίας που περιλαμβάνουν το AI μοντέλο.

Το μοντέλο **Gemini** χρησιμοποιείται για:

- Δημιουργία email καλωσορίσματος.
- Σύνοψη οικονομικών ειδήσεων.
- Παραγωγή προβλεπτικών αναλύσεων.

Στο Σχήμα 4.1 παρουσιάζεται το διάγραμμα ροής λειτουργίας της εφαρμογής, όπου αποτυπώνεται η αλληλεπίδραση μεταξύ χρήστη.



Σχήμα 4.1: Flowchart της ιστοσελίδας. Πηγή: **AI Financial Dashboard**.

4.3 Υλοποίηση Backend

4.3.1 Βάση Δεδομένων και Backend Αποθήκευση

Για την αποθήκευση και διαχείριση των δεδομένων της εφαρμογής χρησιμοποιήθηκε η υπηρεσία **MongoDB Atlas**, μια cloud-based NoSQL βάση δεδομένων. Η επιλογή της συγκεκριμένης τεχνολογίας βασίστηκε στην ευελιξία του μοντέλου εγγράφων (document model), στη δυνατότητα εύκολης κλιμάκωσης, καθώς και στην άμεση ενσωμάτωση με σύγχρονες web εφαρμογές βασισμένες σε JavaScript/TypeScript.

Σε αντίθεση με τις σχεσιακές βάσεις δεδομένων, η MongoDB αποθηκεύει δεδομένα σε μορφή εγγράφων BSON (Binary JSON). Αυτό επιτρέπει τη φυσική αναπαράσταση σύνθετων δομών, όπως πίνακες, εμφωλευμένα αντικείμενα και δυναμικά πεδία, τα οποία ταιριάζουν ιδιαίτερα με τα δεδομένα χρηστών, αγαπημένων στοιχείων, ειδοποιήσεων και ιστορικών ενεργειών που απαιτεί το AI Financial Dashboard.

Η βάση δεδομένων λειτουργεί ως το κεντρικό αποθετήριο πληροφορίας της εφαρμογής και εξυπηρετεί λειτουργίες όπως:

- αποθήκευση στοιχείων χρηστών,
- διαχείριση λίστας αγαπημένων (favorites),
- καταχώρηση ειδοποιήσεων τιμών (alerts),
- διατήρηση ρυθμίσεων και προτιμήσεων χρήστη,
- υποστήριξη των αυτοματοποιημένων ροών που σχετίζονται με emails και AI λειτουργίες.

Η δομή της βάσης οργανώνεται σε **collections** (συλλογές), οι οποίες αντιστοιχούν λειτουργικά σε πίνακες μιας σχεσιακής βάσης. Κάθε collection περιλαμβάνει έγγραφα που περιγράφουν μια συγκεκριμένη οντότητα του συστήματος.

Η βάση δεδομένων της εφαρμογής οργανώνεται σε επιμέρους πίνακες, καθενας από τους οποίους εξυπηρετεί συγκεκριμένο λειτουργικό ρόλο στο σύστημα. Έχει πίνακες όπως:

- **crypto**: Περιλαμβάνει γενικές πληροφορίες για τα κρυπτονομίσματα, όπως αναγνωριστικά, ονομασίες, σύμβολα, εικόνες και τρέχουσες τιμές. Τα δεδομένα αυτά χρησιμοποιούνται κυρίως για την παρουσίαση συνοπτικών στοιχείων στο dashboard.
- **coin_charts**: Αποθηκεύει ιστορικά δεδομένα τιμών κρυπτονομισμάτων σε μορφή χρονοσειρών. Η συλλογή αυτή αποτελεί τη βασική πηγή δεδομένων για τη δημιουργία και απεικόνιση των διαγραμμάτων.
- **stock_news**: Περιέχει τα πιο πρόσφατα οικονομικά νέα που σχετίζονται με μετοχές και αγορές. Τα δεδομένα αυτά χρησιμοποιούνται για την ενημέρωση της αρχικής σελίδας και των προσωποποιημένων ενημερώσεων χρηστών.

- **ai_runs:** Καταγράφει τις ενέργειες ανάλυσης που εκτελούνται μέσω των μοντέλων τεχνητής νοημοσύνης. Για κάθε χρήστη αποθηκεύονται πληροφορίες σχετικά με τα αιτήματα πρόβλεψης ή σύνοψης, επιτρέποντας παρακολούθηση και αξιολόγηση της χρήσης των AI λειτουργιών.
- **users:** Αποτελεί τη βασική συλλογή χρηστών της εφαρμογής. Εκτός από τα στοιχεία ταυτοποίησης, περιλαμβάνει δεδομένα που σχετίζονται με τα αγαπημένα στοιχεία (favorites) και τις ειδοποιήσεις τιμών (alerts) κάθε χρήστη.

4.3.2 Συλλογή crypto

Η συλλογή `crypto` περιλαμβάνει βασικές πληροφορίες για κάθε κρυπτονόμισμα που παρακολουθεί το σύστημα. Κάθε έγγραφο αντιστοιχεί σε ένα συγκεκριμένο νόμισμα και χρησιμοποιείται κυρίως για τη γρήγορη παρουσίαση συνοπτικών οικονομικών στοιχείων στο dashboard.

Ενδεικτική δομή εγγράφου:

- **_id**: Μοναδικό αναγνωριστικό εγγράφου που δημιουργείται αυτόματα από τη MongoDB.
- **id**: Μοναδικό αναγνωριστικό του κρυπτονομίσματος όπως παρέχεται από τον εξωτερικό πάροχο δεδομένων (π.χ. CoinGecko). Χρησιμοποιείται για διασύνδεση με άλλα endpoints και συλλογές.
- **symbol**: Το σύμβολο διαπραγμάτευσης του νομίσματος (ticker), το οποίο εμφανίζεται στο περιβάλλον χρήστη.
- **name**: Η πλήρης ονομασία του κρυπτονομίσματος.
- **image**: Σύνδεσμος προς εικόνα/λογότυπο του νομίσματος, που χρησιμοποιείται για οπτική αναπαράσταση στο UI.
- **current_price**: Η τρέχουσα τιμή του νομίσματος σε δολάρια ΗΠΑ. Το πεδίο αυτό ενημερώνεται περιοδικά.
- **change**: Η ποσοστιαία ή απόλυτη μεταβολή της τιμής σε συγκεκριμένο χρονικό διάστημα, χρησιμοποιούμενη για την ένδειξη ανόδου ή πτώσης.
- **market_cap**: Η συνολική χρηματιστηριακή αξία του νομίσματος, χρήσιμη για συγκρίσεις μεγέθους μεταξύ διαφορετικών assets.
- **last_updated**: Χρονική σήμανση (timestamp) της τελευταίας ενημέρωσης των δεδομένων του νομίσματος.

Πεδίο (Field)	Τύπος Δεδομένων (Type)
<code>_id</code>	ObjectId
<code>id</code>	String
<code>symbol</code>	String
<code>name</code>	String
<code>image</code>	String (URL)
<code>current_price</code>	Number (Double)
<code>change</code>	Number (Double)
<code>market_cap</code>	Number
<code>last_updated</code>	Date

Πίνακας 4.1: Δομή εγγράφου της συλλογής `crypto`

Η συλλογή αυτή λειτουργεί ως η βασική πηγή "ζωντανών" δεδομένων για τα κρυπτονομίσματα και τροφοδοτεί τόσο τις κάρτες τιμών όσο και τα στοιχεία επιλογής νομισμάτων στα διαγράμματα της εφαρμογής.

4.3.3 Συλλογή `coin_charts`

Η συλλογή `coin_charts` αποθηκεύει ιστορικά δεδομένα αγοράς για κάθε κρυπτονόμισμα που παρακολουθεί η εφαρμογή. Τα δεδομένα αυτά χρησιμοποιούνται για τη δημιουργία διαδραστικών διαγραμμάτων τιμών (`price charts`) στο `frontend`, επιτρέποντας στον χρήστη να αναλύει την πορεία ενός `asset` σε βάθος χρόνου.

Κάθε έγγραφο της συλλογής αντιστοιχεί σε ένα συγκεκριμένο νόμισμα και περιλαμβάνει χρονικές σειρές δεδομένων που αφορούν την τιμή και τη συνολική κεφαλαιοποίηση της αγοράς.

- **`coinId`**: Αναγνωριστικό του κρυπτονομίσματος. Συνδέεται με το πεδίο `id` της συλλογής `crypto` και χρησιμοποιείται για την ανάκτηση των σωστών ιστορικών δεδομένων για το επιλεγμένο νόμισμα.
- **`prices`**: Πίνακας (`array`) που περιέχει ιστορικές τιμές του νομίσματος. Κάθε στοιχείο του πίνακα αποτελείται από ένα ζεύγος τιμών της μορφής `[timestamp, price]`, όπου το `timestamp` αναπαριστά τη χρονική στιγμή καταγραφής και το `price` την τιμή του νομίσματος εκείνη τη στιγμή.
- **`market_caps`**: Πίνακας (`array`) που περιέχει ιστορικά δεδομένα κεφαλαιοποίησης αγοράς. Όπως και στο πεδίο `prices`, κάθε στοιχείο έχει τη μορφή `[timestamp, marketCap]`.
- **`updatedAt`**: Χρονική σήμανση (`timestamp`) που δηλώνει την τελευταία φορά που ενημερώθηκαν τα δεδομένα του συγκεκριμένου νομίσματος. Χρησιμοποιείται για έλεγχο φρεσκάδας των δεδομένων.

Πεδίο (Field)	Τύπος Δεδομένων (Type)
<code>coinId</code>	String
<code>prices</code>	Array of <code>[timestamp, price]</code>
<code>market_caps</code>	Array of <code>[timestamp, market cap]</code>
<code>updatedAt</code>	Date

Πίνακας 4.2: Δομή εγγράφου της συλλογής `coin_charts`

4.3.4 Συλλογή users

Η συλλογή `users` περιέχει πληροφορίες για τους εγγεγραμμένους χρήστες της εφαρμογής, τα στοιχεία σύνδεσής τους, καθώς και τις προτιμήσεις τους όπως αγαπημένα στοιχεία και ειδοποιήσεις (alerts). Κάθε έγγραφο αντιστοιχεί σε έναν χρήστη.

- **_id**: Μοναδικό αναγνωριστικό εγγράφου που δημιουργείται αυτόματα από τη MongoDB.
- **name**: Το πλήρες όνομα του χρήστη.
- **email**: Η διεύθυνση ηλεκτρονικού ταχυδρομείου του χρήστη, που χρησιμοποιείται για σύνδεση και ειδοποιήσεις.
- **password**: Το hashed password του χρήστη, αποθηκευμένο με ασφάλεια.
- **role**: Ο ρόλος του χρήστη (π.χ. `user`, `admin`).
- **subscription**: Το είδος συνδρομής που έχει ο χρήστης (π.χ. `free`, `premium`).
- **avatar**: Προαιρετική εικόνα προφίλ του χρήστη (μπορεί να είναι `null`).
- **favorites**: Πίνακας με τα αγαπημένα στοιχεία του χρήστη, όπως κρυπτονομίσματα ή μετοχές.
- **createdAt**: Η ημερομηνία δημιουργίας του λογαριασμού.
- **alerts**: Πίνακας με αλερτς που έχει δημιουργήσει

Πεδίο (Field)	Τύπος Δεδομένων (Type)
<code>_id</code>	<code>ObjectId</code>
<code>name</code>	<code>String</code>
<code>email</code>	<code>String</code>
<code>password</code>	<code>String (hashed)</code>
<code>role</code>	<code>String</code>
<code>subscription</code>	<code>String</code>
<code>avatar</code>	<code>String / Null</code>
<code>favorites</code>	<code>Array</code>
<code>createdAt</code>	<code>Date</code>
<code>alerts</code>	<code>Array</code>

Πίνακας 4.3: Δομή εγγράφου της συλλογής `users`

4.3.5 Συλλογή stock-news

Η συλλογή `stock-news` περιλαμβάνει τα τελευταία νέα σχετικά με εταιρείες, χρηματιστηριακές αγορές και οικονομικές εξελίξεις. Τα δεδομένα χρησιμοποιούνται για την ενημέρωση των χρηστών στο dashboard και για την προβολή headlines και summaries. Κάθε έγγραφο αντιστοιχεί σε μία είδηση.

- **_id**: Μοναδικό αναγνωριστικό εγγράφου που δημιουργείται αυτόματα από τη MongoDB.
- **category**: Κατηγορία της είδησης, π.χ. `company`, `market`, κ.α.
- **datetime**: Χρονική σήμανση της δημοσίευσης σε μορφή Unix timestamp.
- **headline**: Ο τίτλος της είδησης.
- **id**: Αναγνωριστικό της είδησης από την πηγή.
- **image**: Σύνδεσμος προς εικόνα ή λογότυπο που συνοδεύει την είδηση.
- **related**: Το ticker ή το asset που σχετίζεται με την είδηση, π.χ. `AAPL`.
- **source**: Η πηγή προέλευσης της είδησης, π.χ. `Yahoo`.
- **summary**: Περίληψη της είδησης.
- **url**: Σύνδεσμος προς το πλήρες άρθρο.

Πεδίο (Field)	Τύπος Δεδομένων (Type)
<code>_id</code>	ObjectId
<code>category</code>	String
<code>datetime</code>	Number (Unix timestamp)
<code>headline</code>	String
<code>id</code>	Number
<code>image</code>	String (URL)
<code>related</code>	String
<code>source</code>	String
<code>summary</code>	String
<code>url</code>	String (URL)

Πίνακας 4.4: Δομή εγγράφου της συλλογής `stock-news`

4.3.6 Συλλογή ai_runs

Η συλλογή `ai_runs` αποθηκεύει το ιστορικό των αλληλεπιδράσεων των χρηστών με το σύστημα Τεχνητής Νοημοσύνης της εφαρμογής. Κάθε έγγραφο αντιστοιχεί σε μία εκτέλεση (run) ενός AI αιτήματος, όπου ο χρήστης ζητά ανάλυση ή πρόβλεψη για κάποιο χρηματοοικονομικό στοιχείο.

Η συλλογή αυτή επιτρέπει την παρακολούθηση της χρήσης των AI λειτουργιών, την αποθήκευση των αποτελεσμάτων, καθώς και την αναδρομική πρόσβαση σε προηγούμενες αναλύσεις.

- **_id**: Μοναδικό αναγνωριστικό εγγράφου που δημιουργείται αυτόματα από τη MongoDB.
- **userId**: Το αναγνωριστικό του χρήστη που εκτέλεσε το AI αίτημα. Συνδέεται με τη συλλογή `users`.
- **symbol**: Το σύμβολο (ticker) του χρηματοοικονομικού στοιχείου για το οποίο πραγματοποιήθηκε η ανάλυση (π.χ. NFLX).
- **questionType**: Το είδος της ερώτησης ή ανάλυσης που ζητήθηκε από το AI, π.χ. `RISK_ANALYSIS`, `TREND_PREDICTION` κ.α.
- **result**: Το κείμενο αποτελέσματος που παρήχθη από το AI μοντέλο, το οποίο περιλαμβάνει ανάλυση, σύνοψη ή πρόβλεψη.
- **status**: Η κατάσταση της εκτέλεσης (π.χ. `pending`, `completed`, `failed`), χρήσιμη για τον έλεγχο της ροής των ασύγχρονων διεργασιών.
- **createdAt**: Χρονική σήμανση δημιουργίας της εγγραφής, που δείχνει πότε εκτελέστηκε το AI αίτημα.

Πεδίο (Field)	Τύπος Δεδομένων (Type)
<code>_id</code>	ObjectId
<code>userId</code>	ObjectId
<code>symbol</code>	String
<code>questionType</code>	String
<code>result</code>	String (Text)
<code>status</code>	String
<code>createdAt</code>	Date

Πίνακας 4.5: Δομή εγγράφου της συλλογής `ai_runs`

4.4 Υλοποίηση Frontend

4.4.1 Η κεντρική δομή της εφαρμογής — το αρχείο `App.tsx` και `layout.tsx`

Το frontend της εφαρμογής *AI Financial Dashboard* αποτελεί το επίπεδο αλληλεπίδρασης του χρήστη με το σύστημα και σχεδιάστηκε με έμφαση στη διαδραστικότητα, την απόδοση και την

επεκτασιμότητα. Η υλοποίηση βασίστηκε στο σύγχρονο web framework **Next.js** (App Router), αξιοποιώντας τη βιβλιοθήκη **React** για την ανάπτυξη επαναχρησιμοποιήσιμων UI components και την τεχνολογία **TypeScript** για ισχυρή τυποποίηση και αυξημένη αξιοπιστία κώδικα.

Η χρήση του App Router του Next.js επιτρέπει συνδυασμό *Server Components* και *Client Components*, προσφέροντας βελτιστοποιημένη απόδοση, καλύτερη διαχείριση δεδομένων και δυνατότητα server-side επεξεργασίας όπου απαιτείται. Με τον τρόπο αυτό μειώνεται το φορτίο στον client, ενώ παράλληλα βελτιώνεται ο χρόνος φόρτωσης των σελίδων.

Η δομή του frontend βασίζεται σε αρχιτεκτονική βασισμένη σε components, όπου κάθε λειτουργική μονάδα της εφαρμογής (γραφήματα, πίνακες δεδομένων, φόρμες, navigation elements) υλοποιείται ως ανεξάρτητο component. Η προσέγγιση αυτή επιτρέπει:

- Ευκολότερη συντήρηση και επέκταση του κώδικα
- Επαναχρησιμοποίηση στοιχείων διεπαφής
- Καλύτερη οργάνωση της λογικής της εφαρμογής

Για το σχεδιασμό του γραφικού περιβάλλοντος χρησιμοποιήθηκε το **Tailwind CSS**, ένα utility-first framework το οποίο επιτρέπει γρήγορη ανάπτυξη responsive και συνεπούς σχεδιαστικά διεπαφής. Μέσω του Tailwind, το σύστημα επιτυγχάνει προσαρμοστικότητα σε διαφορετικές συσκευές (desktop, tablet, mobile), εξασφαλίζοντας ομοιόμορφη εμπειρία χρήσης.

Η διαχείριση της κατάστασης (state management) στο frontend πραγματοποιείται με χρήση των μηχανισμών της React (hooks), όπως `useState`, `useEffect` και `useContext`, καθώς και με αξιοποίηση των δυνατοτήτων του Next.js για server-side data fetching. Τα δεδομένα που προβάλλονται στα γραφήματα και στις λίστες ειδήσεων αντλούνται μέσω API calls προς τα backend routes, επιτρέποντας δυναμική και σε πραγματικό χρόνο ενημέρωση του περιεχομένου.

Συνολικά, το frontend λειτουργεί ως ένα δυναμικό και ευέλικτο περιβάλλον παρουσίασης δεδομένων, το οποίο συνδέει τον χρήστη με τις δυνατότητες ανάλυσης, προσωποποίησης και τεχνητής νοημοσύνης που προσφέρει το υπόλοιπο σύστημα.

Σε αντίθεση με κλασικές React εφαρμογές που βασίζονται σε ένα κεντρικό αρχείο `App.tsx` για τη δρομολόγηση (routing), η εφαρμογή *AI Financial Dashboard* αξιοποιεί το σύστημα δρομολόγησης του **Next.js App Router**. Στο μοντέλο αυτό, η πλοήγηση δεν ορίζεται χειροκίνητα μέσω ενός κεντρικού component, αλλά προκύπτει από τη δομή των φακέλων στον κατάλογο `app/`. Κάθε φάκελος αντιστοιχεί σε μία διαδρομή (route) της εφαρμογής και κάθε αρχείο `page.tsx` σε μία ξεχωριστή σελίδα.

Η προσέγγιση αυτή επιτρέπει σαφή διαχωρισμό λειτουργικοτήτων, καλύτερη οργάνωση του κώδικα και κλιμακωσιμότητα της εφαρμογής, καθώς νέες σελίδες μπορούν να προστεθούν χωρίς τροποποίηση ενός κεντρικού αρχείου δρομολόγησης. Παράλληλα, το Next.js υποστηρίζει εγγενώς τόσο client-side όσο και server-side πλοήγηση, διατηρώντας την εμπειρία Single

Page Application (SPA), όπου οι μεταβάσεις μεταξύ σελίδων πραγματοποιούνται δυναμικά χωρίς πλήρη ανανέωση του φυλλομετρητή.

Απόσπασμα από main page.tsx: Μας δείχνει πως η react μαζί με την tailwind μας δίνουν την δυνατότητα να γράφουμε καθαρο κωδικα ,component based, reusability.

```
<div className="grid grid-cols-1 lg:grid-cols-[2fr_1.1fr] mt-4">
  <div className="w-full">
    <h3 className="font-semibold text-2xl mb-5">Crypto Overview</h3>
    <PriceCards />
    <CryptoChart />
  </div>
```

Κεντρικό ρόλο στην αρχιτεκτονική του frontend παίζει επίσης το αρχείο layout.tsx. Το layout λειτουργεί ως κοινό «περίβλημα» της εφαρμογής, μέσα στο οποίο ενσωματώνονται σταθερά στοιχεία διεπαφής, όπως η γραμμή πλοήγησης (navigation bar), το γενικό στυλ της σελίδας και η βασική δομή του περιεχομένου. Μέσω του μηχανισμού nested layouts του Next.js, επιτυγχάνεται η επαναχρησιμοποίηση δομικών στοιχείων σε πολλαπλές σελίδες, ενώ παράλληλα επιτρέπεται η ύπαρξη εξειδικευμένων layouts για συγκεκριμένες ενότητες (π.χ. dashboard, profile).

```
<html lang="en" suppressHydrationWarning>
  <body className={` ${inter.className} antialiased `}>
    <GoogleOAuthProvider
      clientId={process.env.NEXT_PUBLIC_GOOGLE_CLIENT_ID!}
    >
      <ThemeProvider
        attribute="class"
        defaultTheme="system"
        enableSystem
        disableTransitionOnChange
      >
        <UserProvider>
          <Navbar initialStocks={initialStocks} />
          <main>
            {children}
            <Toaster />
          </main>
        </UserProvider>
      </ThemeProvider>
    </GoogleOAuthProvider>
  </body>
</html>
```

Σύντομη εξήγηση: Το παραπάνω απόσπασμα αντιπροσωπεύει το βασικό layout της εφαρμογής. Εδώ ορίζονται οι παγκόσμιοι providers όπως το Google OAuth, το ThemeProvider για

διαχείριση θεμάτων (dark/light mode) και ο `UserProvider` για την κατάσταση του χρήστη. Επίσης, περιλαμβάνονται το `Navbar`, το `main content` και το `Toaster` για ειδοποιήσεις, διασφαλίζοντας ότι κάθε σελίδα έχει συνεπή διάταξη και κοινή λειτουργικότητα.

4.4.2 Components

Στη React, τα components αποτελούν τα βασικά δομικά στοιχεία μιας εφαρμογής διεπαφής χρήστη (UI). Ένα component είναι μια ανεξάρτητη, επαναχρησιμοποιήσιμη μονάδα λογικής και παρουσίασης, η οποία περιγράφει πώς πρέπει να εμφανίζεται ένα τμήμα της οθόνης. Η φιλοσοφία αυτή επιτρέπει την κατασκευή πολύπλοκων εφαρμογών μέσω σύνθεσης μικρότερων, διαχειρίσιμων κομματιών.

Κάθε component μπορεί να δέχεται δεδομένα εισόδου (props) και να διατηρεί τη δική του εσωτερική κατάσταση (state). Τα props χρησιμοποιούνται για τη μεταφορά δεδομένων από γονικά σε παιδικά components, ενώ το state επιτρέπει σε ένα component να ανταποκρίνεται δυναμικά σε αλληλεπιδράσεις του χρήστη ή σε αλλαγές δεδομένων.

Σύγχρονες εφαρμογές React βασίζονται κυρίως σε functional components, τα οποία σε συνδυασμό με τα React Hooks (όπως `useState`, `useEffect`) επιτρέπουν τη διαχείριση κατάστασης και κύκλου ζωής με πιο καθαρό και ευανάγνωστο τρόπο.

Παράδειγμα React Component: PriceCards

Το component `PriceCards` αποτελεί ένα χαρακτηριστικό παράδειγμα δυναμικού component που συνδυάζει λήψη δεδομένων από το backend και οπτική απόδοσή τους στο περιβάλλον χρήστη.

Πρωτο τμημα του PriceCard component

```
export default function PriceCards() {
  const [threeTopCurrencies, setThreeTopCurrencies] = useState<
    PriceCardsData[]
  >([]);
  const [loading, setLoading] = useState(true);
  //Now React can mount/unmount all it wants - it will only fetch once per 1
  const didFetch = useRef(false);

  useEffect(() => {
    if (didFetch.current) return;
    didFetch.current = true;

    async function fetchCoins() {
      try {
        const res = await fetch("/api/get-crypto");
        const data: PriceCardsData[] = await res.json();
```

```

const formatData: PriceCardsData[] = (data as unknown[])
  .slice(0, 3)
  .map((coin) => {
    if (
      typeof coin === "object" &&
      coin !== null &&
      "id" in coin &&
      "symbol" in coin &&
      "current_price" in coin &&
      "name" in coin &&
      "change" in coin &&
      "image" in coin
    ) {
      return {
        id: coin.id as string,
        symbol: coin.symbol as string,
        name: coin.name as string,
        price: coin.current_price as number,
        change: `${(coin.change as number).toFixed(2)}%`,
        changeDirection: (coin.change as number) >= 0 ? "up" : "down",
        image: coin.image as string,
      };
    } else {
      throw new Error("Invalid API response structure");
    }
  });
setThreeTopCurrencies(formatData);
} catch (err) {
  console.error(err);
} finally {
  setLoading(false);
}
}
fetchCoins();
}, []);

```

Το πρώτο τμήμα του component είναι υπεύθυνο για την ανάκτηση και επεξεργασία δεδομένων. Αρχικά, χρησιμοποιούνται μεταβλητές κατάστασης (*state*):

- `threeTopCurrencies`: αποθηκεύει τα τρία κρυπτονομίσματα που θα προβληθούν.
- `loading`: δηλώνει αν η εφαρμογή βρίσκεται σε κατάσταση φόρτωσης.

Επιπλέον, χρησιμοποιείται το `useRef` μέσω της μεταβλητής `didFetch`, η οποία λειτουργεί ως μηχανισμός ελέγχου ώστε η κλήση στο API να πραγματοποιείται μόνο μία φορά κατά το πραγματικό φόρτωμα της σελίδας.

Μέσα στο `useEffect`:

1. Πραγματοποιείται έλεγχος ώστε να αποφευχθεί διπλή κλήση του API.
2. Εκτελείται αίτημα προς το endpoint `/api/get-crypto`.
3. Τα δεδομένα που επιστρέφονται υφίστανται μετασχηματισμό (data shaping), ώστε:
 - να κρατηθούν μόνο τα τρία πρώτα στοιχεία,
 - να επιβεβαιωθεί η ορθότητα της δομής τους,
 - να δημιουργηθεί μορφή κατάλληλη για την παρουσίαση στο UI (τιμή, ποσοστό μεταβολής, κατεύθυνση μεταβολής).
4. Τα επεξεργασμένα δεδομένα αποθηκεύονται στο state.
5. Η μεταβλητή `loading` τίθεται σε `false`.

Το τμήμα αυτό υλοποιεί ουσιαστικά τη ροή: *API κλήση* → *έλεγχος δεδομένων* → *μετασχηματισμός* → *αποθήκευση κατάστασης*.

Δευτερο τμημα (return)

```

if (loading) {
  return (
    <div className="grid grid-cols-3 px-5 pt-5 gap-4">
      {"123".split("").map((_, key) => (
        <Skeleton key={key} className="h-[85px] rounded-xl" />
      ))}
    </div>
  );
}

return (
  <div className="grid gap-4 md:grid-cols-3 px-5 pt-5">
    {threeTopCurrencies?.map((crypto: PriceCardsData, index: number) => (
      <Card key={index} className="shadow-none border-none p-4 w-full">
        <div className="flex gap-3">
          {/* IMAGE */}
          <img
            src={crypto.image}
            alt={crypto.name}
            className="h-11 w-11 object-contain rounded-md"
          />

          {/* RIGHT SIDE BLOCK */}
          <div className="flex flex-col justify-center">
            {/* NAME */}

```

```

<p className="text-xs font-medium text-muted-foreground">
  {crypto.name}
</p>

{/* PRICE + ARROW (same row) */}
<div className="flex items-center justify-between w-full">
  {/* PRICE */}
  <p className="text-xl font-bold">
    {new Intl.NumberFormat("en-US", {
      style: "currency",
      currency: "USD",
    }).format(crypto.price)}
  </p>

  {/* ARROW + CHANGE */}
  <div
    className={`flex items-center gap-1 ml-4 ${
      crypto.changeDirection === "up"
        ? "text-green-500"
        : "text-red-500"
    }`}
  >
    {crypto.changeDirection === "down" ? (
      <ArrowBigDown className="text-xl" />
    ) : (
      <ArrowBigDown className="text-xl rotate-180" />
    )}
    <span className="text-sm">{crypto.change}</span>
  </div>
</div>
</div>
</div>
</Card>
  )}
</div>
);

```

Το δεύτερο τμήμα του component αφορά την οπτική απόδοση των δεδομένων.

Αρχικά, ελέγχεται αν η εφαρμογή βρίσκεται σε κατάσταση φόρτωσης. Σε αυτή την περίπτωση εμφανίζονται *skeleton components*, τα οποία λειτουργούν ως προσωρινά placeholders βελτιώνοντας την εμπειρία χρήστη.

Όταν τα δεδομένα είναι διαθέσιμα, γίνεται επανάληψη (map) στον πίνακα `threeTopCurrencies`. Για κάθε κρυπτονόμισμα δημιουργείται μία κάρτα που περιλαμβάνει:

- εικόνα του νομίσματος,

- όνομα,
- τρέχουσα τιμή μορφοποιημένη σε δολάρια,
- ένδειξη μεταβολής τιμής με ποσοστό και εικονίδιο κατεύθυνσης.

Το χρώμα και το εικονίδιο μεταβάλλονται δυναμικά ανάλογα με το αν η τιμή έχει αυξηθεί ή μειωθεί, προσφέροντας άμεση οπτική πληροφόρηση.

Συνολικά, το component `PriceCards` αποτελεί παράδειγμα *data-driven UI*, όπου τα δεδομένα από το backend καθορίζουν πλήρως τη δομή και τη συμπεριφορά της διεπαφής.

4.4.3 Παράδειγμα React Component: CryptoChart

Το component `CryptoChart` είναι υπεύθυνο για την οπτικοποίηση της ιστορικής πορείας τιμής κρυπτονομισμάτων μέσω διαγράμματος. Αποτελεί παράδειγμα component ανάλυσης δεδομένων, στο οποίο συνδυάζονται διαχείριση κατάστασης, επιλογές χρήστη και επεξεργασία χρονοσειρών.

```
export function CryptoChart() {
  const [loading, setLoading] = useState(true);
  const [error, setError] = useState(false);

  const [value, setValue] = useState<string>("");
  const [formattedPrice, setFormattedPrice] = useState("");
  const [selectedPeriod, setSelectedPeriod] = useState("7D");
  const [comboBoxCoins, setComboBoxCoins] = useState<CryptoComboBox[]>([]);
  const selectedCoin = comboBoxCoins?.find((coin) => coin.value === value);

  const [chartData, setChartData] = useState<chartData[]>([]);

  const didFetch = useRef(false);

  function aggregateByDay(data: [number, number][], days: number) {
    if (!data.length) return [];

    const maxTs = Math.max(...data.map((d) => d[0]));
    const cutoff = maxTs - days * 86400000;

    const map: Record<string, number> = {};

    for (const [ts, raw] of data) {
      if (ts < cutoff) continue;

      const price = Number(raw);
```

```

    if (isNaN(price)) continue;

    const day = new Date(ts).toLocaleDateString("en-US", {
      timeZone: "UTC",
      month: "short",
      day: "numeric",
    });

    map[day] = price;
  }

  return Object.entries(map).map(([date, price]) => ({ date, price }));
}

```

Διαχείριση Κατάστασης (State Management)

Στην αρχή του component ορίζονται πολλαπλές μεταβλητές κατάστασης:

- `loading`: δηλώνει αν τα δεδομένα του γραφήματος βρίσκονται σε διαδικασία φόρτωσης.
- `error`: υποδεικνύει πιθανό σφάλμα κατά την ανάκτηση δεδομένων.
- `value`: αποθηκεύει την τιμή που έχει επιλέξει ο χρήστης από το *combo box* (επιλεγμένο κρυπτονόμισμα).
- `formattedPrice`: περιέχει την τρέχουσα τιμή του επιλεγμένου νομίσματος σε μορφοποιημένη μορφή.
- `selectedPeriod`: δηλώνει το χρονικό διάστημα απεικόνισης (π.χ. 7 ημέρες).
- `comboBoxCoins`: πίνακας με τα διαθέσιμα κρυπτονομίσματα προς επιλογή.
- `chartData`: τα τελικά δεδομένα που θα χρησιμοποιηθούν για τη δημιουργία του γραφήματος.

Η μεταβλητή:

```
const selectedCoin = comboBoxCoins.find((coin) => coin.value === value);
```

επιτρέπει τον εντοπισμό του αντικειμένου που αντιστοιχεί στο κρυπτονόμισμα που έχει επιλέξει ο χρήστης.

Παράλληλα, το `useRef` με τη μεταβλητή `didFetch` χρησιμοποιείται για τον έλεγχο ώστε τα αρχικά δεδομένα να φορτώνονται μόνο μία φορά.

Επεξεργασία Δεδομένων Χρονοσειράς

Η συνάρτηση `aggregateByDay` είναι υπεύθυνη για τον μετασχηματισμό των πρωτογενών δεδομένων τιμών σε μορφή κατάλληλη για γραφική απεικόνιση.

Δέχεται:

- έναν πίνακα ζευγών `[timestamp, price]`,
- τον αριθμό ημερών που πρέπει να εμφανιστούν στο γράφημα.

Η λειτουργία της συνοψίζεται στα εξής βήματα:

1. Εντοπίζει το πιο πρόσφατο χρονικό στιγμιότυπο και υπολογίζει το κατώφλι (cutoff) βάσει του επιλεγμένου πλήθους ημερών.
2. Φιλτράρει τα δεδομένα ώστε να κρατηθούν μόνο όσα ανήκουν στο επιθυμητό χρονικό εύρος.
3. Μετατρέπει τις χρονικές σφραγίδες σε ημερομηνίες (UTC μορφή).
4. Ομαδοποιεί τις τιμές ανά ημέρα και κρατά την τελευταία διαθέσιμη τιμή για κάθε ημέρα.

Το αποτέλεσμα είναι ένας πίνακας αντικειμένων της μορφής:

```
{ date, price }
```

που χρησιμοποιείται άμεσα από το σύστημα γραφημάτων.

Η συνάρτηση αυτή αποτελεί βασικό βήμα προεπεξεργασίας, καθώς μετατρέπει μια ακατέργαστη χρονοσειρά σε δομή κατανοητή και οπτικά αξιοποιήσιμη από τη διεπαφή χρήστη.

Δευτερο κομματι

```
useEffect(() => {
  if (didFetch.current) return;
  didFetch.current = true;

  async function fetchCoins() {
    try {
      const res = await fetch("/api/get-crypto");
      const data = await res.json();

      const formattedData: CryptoComboBox[] = (data as unknown[])
        .map((crypto: unknown) => {
          if (
            typeof crypto === "object" &&
            crypto !== null &&
            "id" in crypto &&
            "name" in crypto &&
            "image" in crypto &&

```

```

        "current_price" in crypto &&
        "change" in crypto
    ) {
        return {
            value: crypto.id as string,
            label: crypto.name as string,
            icon: crypto.image as string,
            price: String(crypto.current_price),
            change:
                crypto.change != null
                ? (crypto.change as number).toFixed(4)[0] !== "-"
                ? `+${(crypto.change as number).toFixed(4)}\`
                : (crypto.change as number).toFixed(4)
                : "0.0000",
        };
    } else {
        return null; //return null from invalid entries
    }
}
})
.filter((item): item is CryptoComboBox => item !== null);
setComboBoxCoins(formattedData);
} catch (err) {
    setError(true);
    console.error(err);
} finally {
    setLoading(false);
}
}
fetchCoins();
}, []);

//Automatically select the first coin when data is available
useEffect(() => {
    if (comboBoxCoins && comboBoxCoins.length > 0 && !value)
    {
        setValue(comboBoxCoins[0].value);
    }
}, [comboBoxCoins, value]);

```

Αρχική Ανάκτηση και Προετοιμασία Δεδομένων

Στο επόμενο στάδιο του component χρησιμοποιείται το `useEffect` για την αρχική φόρτωση των διαθέσιμων κρυπτονομισμάτων που θα εμφανίζονται στο πεδίο επιλογής (combo box). Αρχικά, μέσω της μεταβλητής `didFetch` ελέγχεται ώστε η διαδικασία ανάκτησης να εκτελείται μόνο μία φορά:

- Πραγματοποιείται κλήση στο endpoint `/api/get-crypto`.
- Τα δεδομένα που επιστρέφονται μετατρέπονται σε πίνακα αντικειμένων τύπου `CryptoComboBox`.

Κατά τον μετασχηματισμό:

1. Ελέγχεται η ορθότητα της δομής κάθε αντικειμένου (ύπαρξη πεδίων `id`, `name`, `image`, `current_price`, `change`).
2. Δημιουργείται νέο αντικείμενο που περιλαμβάνει:
 - `value`: αναγνωριστικό νομίσματος,
 - `label`: όνομα νομίσματος,
 - `icon`: εικόνα,
 - `price`: τρέχουσα τιμή σε μορφή συμβολοσειράς,
 - `change`: μορφοποιημένη μεταβολή τιμής με πρόσημο.
3. Τυχόν μη έγκυρες εγγραφές απορρίπτονται.

Στη συνέχεια, τα επεξεργασμένα δεδομένα αποθηκεύονται στο `state comboBoxCoins`. Σε περίπτωση αποτυχίας της κλήσης, ενεργοποιείται η κατάσταση σφάλματος (`error`), ενώ με την ολοκλήρωση της διαδικασίας η μεταβλητή `loading` τίθεται σε `false`.

Αυτόματη Επιλογή Νομίσματος

Ένα δεύτερο `useEffect` χρησιμοποιείται για τη βελτίωση της εμπειρίας χρήστη. Όταν τα δεδομένα των κρυπτονομισμάτων φορτωθούν επιτυχώς και δεν έχει ήδη επιλεγεί κάποιο νόμισμα, το σύστημα επιλέγει αυτόματα το πρώτο διαθέσιμο στοιχείο της λίστας:

```
setValue (comboBoxCoins [0] .value)
```

Με τον τρόπο αυτό, το γράφημα μπορεί να εμφανιστεί άμεσα χωρίς να απαιτείται αρχική ενέργεια από τον χρήστη.

Φορτώση ιστορικών δεδομένων

```
useEffect (() => {
  async function fetchPrices () {
    if (selectedCoin?.value) {
      console.log ("fetching chart for:", selectedCoin?.value);
      const res = await fetch (`/api/get-crypto-chart/${selectedCoin.value}`);
      const data = await res.json ();

      if (!res.ok) {
        console.error ("Error loading chart:", data.error);
        return;
      }
    }
  }
})
```

```

const prices: [number, number][] = data.prices;
console.log("RAW API prices:", prices.slice(0, 5));

const filteredPrices = aggregateByDay(
  prices,
  selectedPeriod === "7D" ? 7 : selectedPeriod === "15D" ? 15 : 30
);
console.log("FINAL chartData:", filteredPrices);
setChartData(filteredPrices);
}
}
fetchPrices();
}, [selectedCoin, selectedPeriod]);

```

Δυναμική Ανάκτηση Δεδομένων Γραφήματος

Το τελικό `useEffect` του `component` είναι υπεύθυνο για τη φόρτωση των ιστορικών δεδομένων τιμής που θα χρησιμοποιηθούν στο γράφημα. Η εκτέλεσή του εξαρτάται από δύο παραμέτρους: το επιλεγμένο κρυπτονόμισμα (`selectedCoin`) και το χρονικό διάστημα απεικόνισης (`selectedPeriod`).

Κάθε φορά που κάποια από αυτές τις τιμές αλλάζει, εκτελείται η συνάρτηση `fetchPrices`.

1. Ελέγχεται αν υπάρχει έγκυρο επιλεγμένο νόμισμα.
2. Πραγματοποιείται αίτημα προς το `endpoint`:

```
/api/get-crypto-chart/id
```

όπου `id` είναι το αναγνωριστικό του νομίσματος.

3. Αν η απόκριση περιέχει σφάλμα, η διαδικασία διακόπτεται.
4. Τα δεδομένα που επιστρέφονται είναι χρονοσειρά τιμών της μορφής:

```
[timestamp, price]
```

Στη συνέχεια υφίστανται επεξεργασία μέσω της συνάρτησης `aggregateByDay`, η οποία:

- φιλτράρει τις τιμές βάσει του επιλεγμένου χρονικού εύρους (7, 15 ή 30 ημέρες),
- ομαδοποιεί τις τιμές ανά ημέρα,
- επιστρέφει δεδομένα κατάλληλα για οπτικοποίηση.

Τέλος, τα επεξεργασμένα δεδομένα αποθηκεύονται στη μεταβλητή κατάστασης `chartData`, η οποία αποτελεί την κύρια πηγή δεδομένων για το σύστημα γραφημάτων της διεπαφής.

Απόδοση Διεπαφής

Ως τελικό στάδιο εκτέλεσης του component, πραγματοποιείται η εντολή `return`, μέσω της οποίας αποδίδεται το γραφικό περιβάλλον του διαγράμματος. Στο στάδιο αυτό γίνεται η οπτική παρουσίαση των δεδομένων που έχουν ήδη ανακτηθεί και επεξεργαστεί, ολοκληρώνοντας τη ροή λειτουργίας του component από τη λήψη δεδομένων έως την εμφάνισή τους στη διεπαφή χρήστη.

Κεφάλαιο 5

Παρουσιαση εφαρμογης AI Financial Dashbaord

5.1 Αρχική Σελίδα (Dashboard)

Ανοίγοντας την εφαρμογή, ο χρήστης μεταφέρεται στην αρχική σελίδα (Dashboard), η οποία αποτελεί το κεντρικό σημείο πληροφόρησης της πλατφόρμας. Η σελίδα αυτή έχει σχεδιαστεί ώστε να παρέχει άμεσα μια συνοπτική αλλά ουσιαστική εικόνα της τρέχουσας κατάστασης των αγορών.

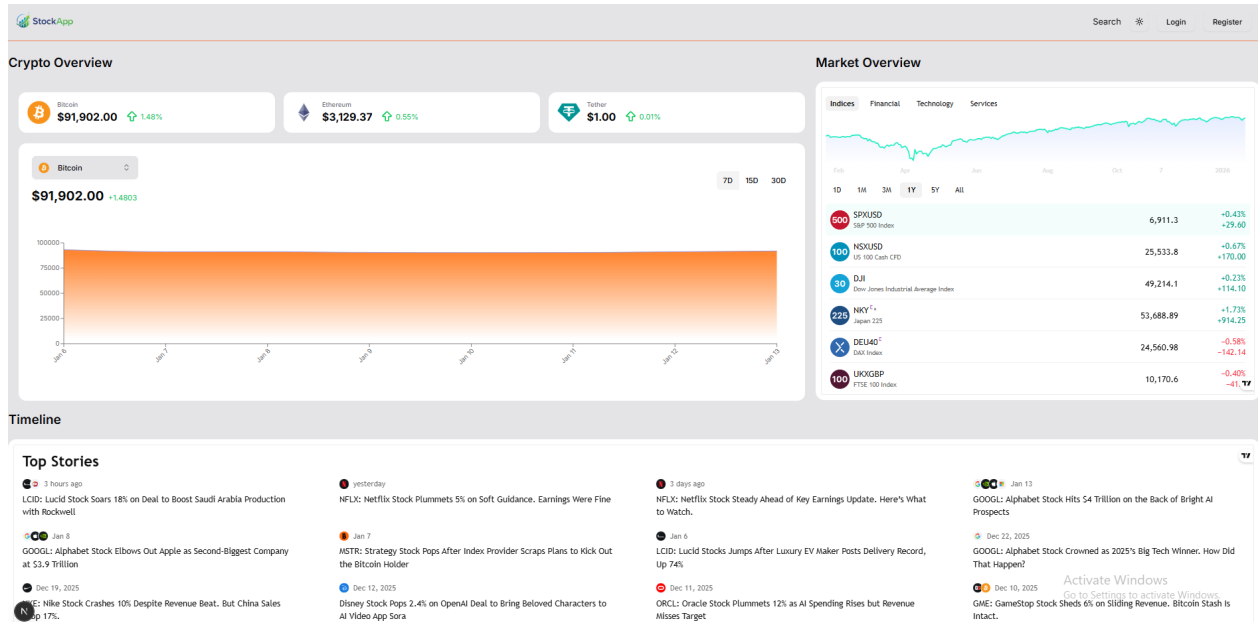
Στο επάνω μέρος της διεπαφής εμφανίζονται βασικά στοιχεία πλοήγησης, επιτρέποντας στον χρήστη να μεταβεί σε επιμέρους λειτουργίες της εφαρμογής, όπως η διαχείριση αγαπημένων στοιχείων, οι ειδοποιήσεις (alerts) και οι λειτουργίες ανάλυσης με Τεχνητή Νοημοσύνη.

Στο κύριο τμήμα του dashboard παρουσιάζονται διαδραστικά γραφήματα που αφορούν τόσο την αγορά κρυπτονομισμάτων όσο και βασικούς χρηματιστηριακούς τομείς (π.χ. indices, financial, technology, services). Παράλληλα, προβάλλονται τα πιο πρόσφατα οικονομικά νέα, προσφέροντας στον χρήστη μια ολοκληρωμένη και άμεση ενημέρωση.

Συγκεκριμένα, στο Dashboard ο χρήστης μπορεί να βρει:

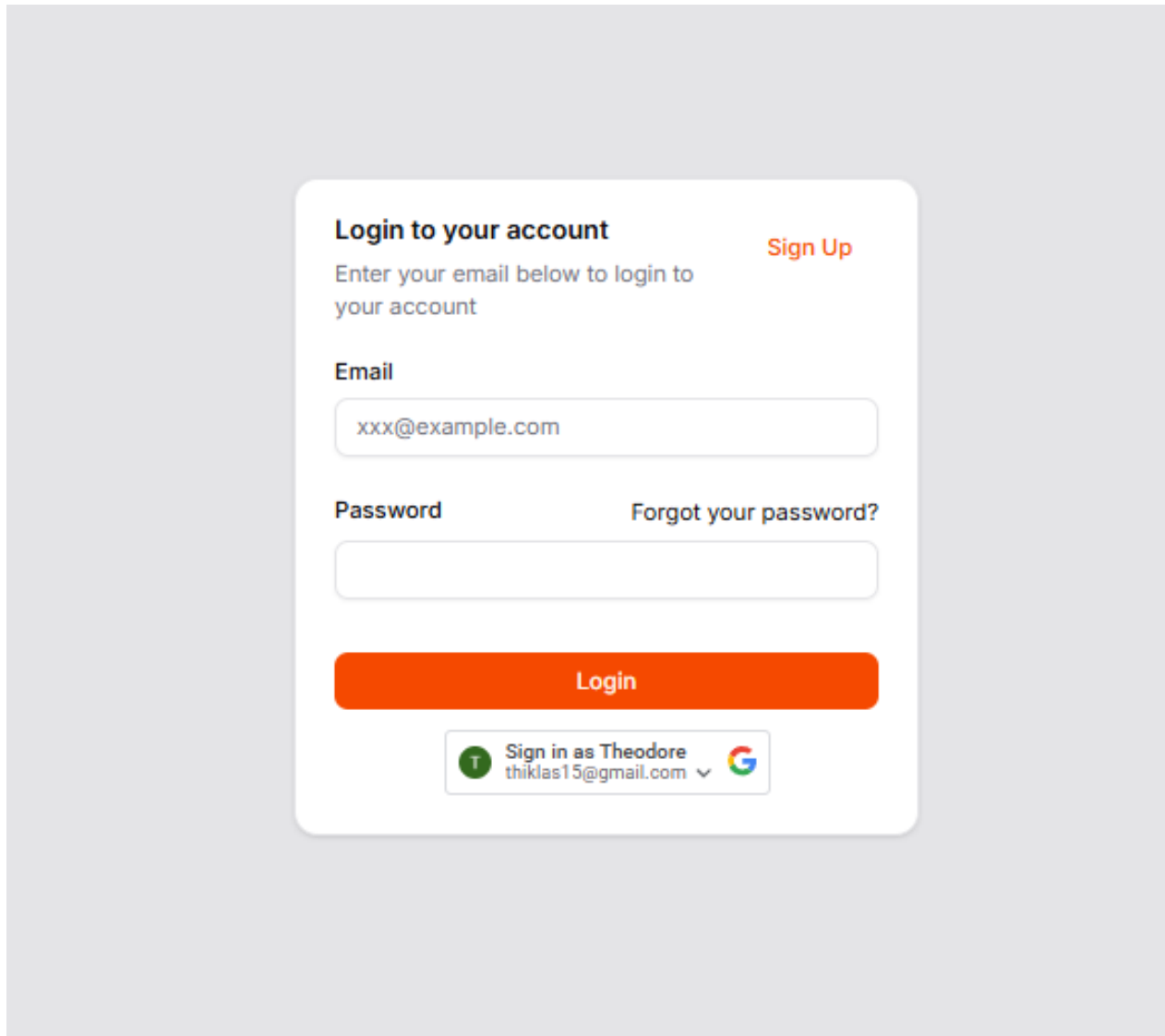
- **Γράφημα Κρυπτονομισμάτων:** Διαδραστικό γράφημα που απεικονίζει την πορεία επιλεγμένων κρυπτονομισμάτων. Ο χρήστης μπορεί να αλλάξει το νόμισμα προβολής καθώς και το χρονικό εύρος (π.χ. 7, 15 ή 30 ημέρες), ώστε να αναλύει βραχυπρόθεσμες ή μεσοπρόθεσμες τάσεις.
- **Γράφημα Χρηματιστηριακών Τομέων:** Παρουσιάζει δεδομένα από βασικούς χρηματιστηριακούς τομείς, όπως indices, financial, technology και services. Το γράφημα επιτρέπει σύγκριση μεταξύ διαφορετικών κατηγοριών της αγοράς.
- **Κάρτες Τιμών (Price Cards):** Συνοπτική παρουσίαση βασικών στοιχείων, όπως τρέχουσα τιμή και μεταβολή, για επιλεγμένα δημοφιλή κρυπτονομίσματα ή μετοχές. Οι κάρτες αυτές λειτουργούν ως γρήγορη ένδειξη ανόδου ή πτώσης της αγοράς.
- **Τελευταία Οικονομικά Νέα:** Εμφάνιση πρόσφατων ειδήσεων που σχετίζονται με αγορές και εταιρείες. Οι ειδήσεις συνοδεύονται από τίτλο, εικόνα και σύντομη περίληψη, επιτρέποντας στον χρήστη να ενημερώνεται άμεσα για σημαντικά γεγονότα.

- **Δυνατότητα Πρόσβασης χωρίς Σύνδεση:** Το βασικό περιεχόμενο του Dashboard είναι διαθέσιμο και σε μη συνδεδεμένους χρήστες, δίνοντας τη δυνατότητα σε οποιονδήποτε να δει δεδομένα αγοράς και γενικές πληροφορίες.



Σχήμα 5.1: Αρχική σελίδα τις ιστοσελίδας. Πηγή: **AI Financial Dashboard**.

Δυνατότητα Σύνδεσης και Δημιουργίας Λογαριασμού: Από τη γραμμή πλοήγησης (Navbar), ο επισκέπτης έχει τη δυνατότητα είτε να συνδεθεί σε υπάρχον λογαριασμό είτε να δημιουργήσει νέο. Η διαδικασία αυτή αποτελεί το σημείο μετάβασης από τη γενική, δημόσια λειτουργικότητα της εφαρμογής στις εξατομικευμένες υπηρεσίες. Μέσω της σύνδεσης, ενεργοποιούνται επιπλέον δυνατότητες, όπως η αποθήκευση αγαπημένων στοιχείων, η δημιουργία ειδοποιήσεων (alerts), η λήψη προσωποποιημένων ενημερώσεων μέσω email και η χρήση των λειτουργιών ανάλυσης με Τεχνητή Νοημοσύνη.



Σχήμα 5.2: Σελίδα login-register. Πηγή: [AI Financial Dashboard](#).

5.2 Διαχείριση Αγαπημένων Στοιχείων (Manage Favorites)

Εφόσον ο χρήστης συνδεθεί στην εφαρμογή, αποκτά πρόσβαση σε επιπλέον λειτουργίες που αφορούν την προσωποποιημένη διαχείριση των χρηματοοικονομικών στοιχείων που τον ενδιαφέρουν. Μία από τις βασικότερες λειτουργίες είναι η ενότητα *Manage Favorites*.

Η σελίδα αυτή επιτρέπει στον χρήστη να διαχειρίζεται τη λίστα των αγαπημένων του στοιχείων (κρυπτονομίσματα και μετοχές), τα οποία έχει επιλέξει για ταχύτερη πρόσβαση και εξατομικευμένη παρακολούθηση.

Συγκεκριμένα, ο χρήστης μπορεί:

- **Να προβάλλει όλα τα αγαπημένα του στοιχεία:** Παρουσιάζεται συγκεντρωτική λίστα με τα assets που έχει προσθέσει στα favorites, μαζί με βασικές πληροφορίες όπως όνομα, σύμβολο και τρέχουσα κατάσταση τιμής.
- **Να προσθέτει νέα στοιχεία:** Ο χρήστης μπορεί να επιλέξει οποιοδήποτε διαθέσιμο κρυπτονόμισμα ή μετοχή και να το προσθέσει στη λίστα αγαπημένων του, ώστε να το παρακολουθεί πιο άμεσα.
- **Να αφαιρεί στοιχεία από τα αγαπημένα:** Παρέχεται η δυνατότητα διαγραφής ενός στοιχείου από τη λίστα favorites, εφόσον ο χρήστης δεν επιθυμεί πλέον να το παρακολουθεί.
- **Να αποκτά γρήγορη πρόσβαση σε σχετικές λειτουργίες:** Μέσα από τη σελίδα διαχείρισης, ο χρήστης μπορεί να μεταβεί σε περαιτέρω ενέργειες, όπως η δημιουργία ειδοποιήσεων (alerts) ή η εκτέλεση αναλύσεων μέσω AI για συγκεκριμένα αγαπημένα στοιχεία.

Η λειτουργία των favorites αποτελεί βασικό μηχανισμό προσωποποίησης της εφαρμογής, καθώς επιτρέπει την προσαρμογή του περιεχομένου και των ειδοποιήσεων στις προτιμήσεις κάθε χρήστη.

5.3 Λειτουργίες Σελίδας Manage Favorites

Η σελίδα *Manage Favorites* αποτελεί το βασικό σημείο διαχείρισης των αγαπημένων χρηματοοικονομικών στοιχείων του χρήστη. Σε αυτήν συγκεντρώνονται όλες οι σχετικές πληροφορίες και ενέργειες που αφορούν τα assets που έχει επιλέξει να παρακολουθεί.

Company	Symbol	Price	Change	Market Cap	P/E Ratio	Alert	Action
Apple Inc	AAPL	\$259.48	+0.46%	\$3.81T	32.5	Add Alert	🟡
Netflix Inc	NFLX	\$83.49	+0.40%	\$352.51B	35.8	Add Alert	🟡
Microsoft Corp	MSFT	\$430.29	-0.74%	\$3.20T	34.0	Add Alert	🟡

Σχήμα 5.3: Σελίδα watchlist. Πηγή: [AI Financial Dashboard](#).

5.3.1 Πίνακας Αγαπημένων Στοιχείων (Watchlist Table)

Κεντρικό στοιχείο της σελίδας είναι ο πίνακας αγαπημένων στοιχείων (watchlist table). Ο πίνακας αυτός παρουσιάζει συγκεντρωτικά όλα τα κρυπτονομίσματα και τις μετοχές που έχει προσθέσει ο χρήστης στα favorites του.

Ο σκοπός του πίνακα είναι να προσφέρει μια οργανωμένη και άμεση επισκόπηση των στοιχείων ενδιαφέροντος, λειτουργώντας ως προσωπικό “monitor” αγοράς.

Για κάθε στοιχείο εμφανίζονται βασικές πληροφορίες, όπως:

- **Όνομα και σύμβολο (name, ticker):** Επιτρέπουν την άμεση αναγνώριση του στοιχείου.
- **Τρέχουσα τιμή:** Παρουσιάζει την πιο πρόσφατη διαθέσιμη τιμή, όπως έχει ανακτηθεί από το backend.
- **Μεταβολή τιμής:** Οπτική ένδειξη ανόδου ή πτώσης (π.χ. θετική ή αρνητική μεταβολή), βοηθώντας τον χρήστη να εντοπίσει γρήγορα τις τάσεις.
- **Market Cap (Κεφαλαιοποίηση Αγοράς):** Εκφράζει τη συνολική χρηματιστηριακή αξία του στοιχείου και χρησιμοποιείται για την εκτίμηση του μεγέθους και της θέσης του στην αγορά. Δίνει στον χρήστη μια ένδειξη για το αν πρόκειται για μεγάλο, μεσαίο ή μικρό asset.
- **P/E Ratio (Price-to-Earnings):** Δείκτης που χρησιμοποιείται κυρίως για μετοχές και αποτυπώνει τη σχέση μεταξύ της τιμής της μετοχής και των κερδών της εταιρείας. Βοηθά τον χρήστη να αξιολογήσει αν μια μετοχή θεωρείται σχετικά υπερτιμημένη ή υποτιμημένη.
- **Ενέργειες (Actions):** Για κάθε γραμμή του πίνακα, ο χρήστης μπορεί να εκτελέσει επιπλέον λειτουργίες, όπως αφαίρεση από τα αγαπημένα, δημιουργία ειδοποίησης (alert) ή μετάβαση σε AI ανάλυση για το συγκεκριμένο στοιχείο.

Ο πίνακας ενημερώνεται δυναμικά μέσω κλήσεων στο backend, διασφαλίζοντας ότι τα δεδομένα που προβάλλονται είναι όσο το δυνατόν πιο πρόσφατα. Με τον τρόπο αυτό, ο χρήστης διαθέτει ένα εξατομικευμένο περιβάλλον παρακολούθησης της αγοράς, προσαρμοσμένο στις δικές του επιλογές.

5.3.2 Πίνακας Ειδοποιήσεων (Alerts Table)

Δίπλα από τον πίνακα αγαπημένων στοιχείων εμφανίζεται ο πίνακας ειδοποιήσεων (alerts table). Ο πίνακας αυτός παρουσιάζει συγκεντρωτικά όλες τις ειδοποιήσεις που έχει δημιουργήσει ο χρήστης, επιτρέποντάς του να παρακολουθεί και να διαχειρίζεται τις αυτοματοποιημένες συνθήκες που έχει ορίσει.

Κάθε εγγραφή του πίνακα αντιστοιχεί σε μία ειδοποίηση που συνδέεται με συγκεκριμένο χρηματοοικονομικό στοιχείο (κρυπτονόμισμα ή μετοχή) και βασίζεται σε κανόνες που έχει θέσει ο ίδιος ο χρήστης.

Για κάθε ειδοποίηση εμφανίζονται πληροφορίες όπως:

- **Στοιχείο (Asset):** Το όνομα ή το σύμβολο του asset στο οποίο αναφέρεται η ειδοποίηση.
- **Συνθήκη (Condition):** Ο λογικός κανόνας ενεργοποίησης, π.χ. $price < 50$ ή $price > 200$. Η συνθήκη αυτή ελέγχεται περιοδικά από το backend.
- **Κατάσταση (Status):** Ένδειξη για το αν η ειδοποίηση είναι ενεργή, έχει ενεργοποιηθεί ή έχει απενεργοποιηθεί.

Δυνατότητες Διαχείρισης Ειδοποιήσεων

Ο χρήστης μπορεί να αλληλεπιδρά με κάθε ειδοποίηση μέσω ενεργειών που παρέχονται στον πίνακα:

- **Δημιουργία νέας ειδοποίησης:** Ο χρήστης μπορεί να ορίσει νέα συνθήκη τιμής για κάποιο αγαπημένο στοιχείο.
- **Επεξεργασία ειδοποίησης (Edit):** Δίνεται η δυνατότητα τροποποίησης της συνθήκης ή του στοιχείου στο οποίο αναφέρεται η ειδοποίηση, χωρίς να απαιτείται διαγραφή και εκ νέου δημιουργία.
- **Διαγραφή ειδοποίησης (Delete):** Ο χρήστης μπορεί να αφαιρέσει πλήρως μια ειδοποίηση από το σύστημα.
- **Αυτόματη ενεργοποίηση συμβάντος:** Όταν η συνθήκη μιας ειδοποίησης ικανοποιηθεί, δημιουργείται ένα συμβάν στο backend, το οποίο οδηγεί στην αποστολή αναλυτικού email ειδοποίησης προς τον χρήστη. Η διαδικασία αυτή βασίζεται σε event-driven λογική εκτέλεσης η οποία εκτελείται μια φορά την ημέρα.

Ο πίνακας ειδοποιήσεων λειτουργεί ως μηχανισμός αυτοματοποιημένης παρακολούθησης της αγοράς, επιτρέποντας στον χρήστη να ενημερώνεται άμεσα για σημαντικές μεταβολές, χωρίς να απαιτείται συνεχής χειροκίνητος έλεγχος των τιμών.

5.3.3 Ενότητα Ανάλυσης με Τεχνητή Νοημοσύνη (AI Analysis Section)

Κάτω από τους πίνακες αγαπημένων στοιχείων και ειδοποιήσεων, παρέχεται η ενότητα ανάλυσης με Τεχνητή Νοημοσύνη, η οποία επιτρέπει στον χρήστη να αλληλεπιδρά δυναμικά με το σύστημα μέσω προκαθορισμένων ερωτήσεων ανάλυσης.

Τρόπος Λειτουργίας

Ο χρήστης επιλέγει ένα από τα αγαπημένα του στοιχεία και στη συνέχεια μπορεί να υποβάλει ερωτήσεις ανάλυσης μέσω έτοιμων επιλογών (predefined questions), όπως:

- Ανάλυση ρίσκου (Risk Analysis)
- Εκτίμηση τάσης (Trend Analysis)
- Γενική σύνοψη κατάστασης του asset

Με την υποβολή του αιτήματος:

1. Το frontend αποστέλλει το ερώτημα στο backend μαζί με το επιλεγμένο σύμβολο.
2. Το backend ανακτά σχετικά οικονομικά δεδομένα.

3. Τα δεδομένα αποστέλλονται στο AI μοντέλο (Gemini), το οποίο παράγει φυσικής γλώσσας ανάλυση.
4. Το αποτέλεσμα επιστρέφεται στην εφαρμογή και εμφανίζεται στον χρήστη σε μορφή κειμένου.

Αποθήκευση Ιστορικού

Κάθε εκτέλεση AI ανάλυσης καταγράφεται στη συλλογή `ai_runs` της βάσης δεδομένων. Με τον τρόπο αυτό:

- διατηρείται ιστορικό των ερωτημάτων του χρήστη,
- επιτρέπεται η αναδρομή σε προηγούμενα αποτελέσματα,
- εξασφαλίζεται ιχνηλασιμότητα των AI λειτουργιών.

5.3.4 Ημερήσιες Ενημερώσεις (Daily News)

Η εφαρμογή παρέχει τη δυνατότητα αποστολής ημερήσιων ενημερωτικών emails (Daily News) στους συνδεδεμένους χρήστες, με στόχο τη συνεχή και αυτοματοποιημένη ενημέρωσή τους για τις εξελίξεις που αφορούν τα αγαπημένα τους χρηματοοικονομικά στοιχεία.

Η λειτουργία αυτή βασίζεται στη σύνδεση των αγαπημένων στοιχείων (favorites) με ειδησεογραφικές πηγές και αξιοποιεί τόσο μηχανισμούς ανάκτησης δεδομένων όσο και Τεχνητή Νοημοσύνη για την επεξεργασία του περιεχομένου.

Ροή Λειτουργίας

Η διαδικασία αποστολής των ημερήσιων ειδήσεων ακολουθεί τα παρακάτω βήματα:

1. Το σύστημα ανακτά τις πιο πρόσφατες ειδήσεις που σχετίζονται με τα αγαπημένα στοιχεία κάθε χρήστη.
2. Οι ειδήσεις αποθηκεύονται προσωρινά.
3. Το περιεχόμενο των ειδήσεων αποστέλλεται στο AI μοντέλο (Gemini), το οποίο παράγει συνοπτικές περιλήψεις (summaries) σε φυσική γλώσσα.
4. Δημιουργείται προσωποποιημένο email, το οποίο περιλαμβάνει τις περιλήψεις των σημαντικότερων ειδήσεων.
5. Το email αποστέλλεται αυτόματα στον χρήστη.

Fallback Μηχανισμός

Σε περίπτωση που ο χρήστης δεν έχει προσθέσει κανένα στοιχείο στα αγαπημένα του, το σύστημα ενεργοποιεί μηχανισμό fallback. Σε αυτή την περίπτωση, αποστέλλονται γενικές οικονομικές και χρηματιστηριακές ειδήσεις, διασφαλίζοντας ότι ο χρήστης λαμβάνει ενημέρωση ακόμη και χωρίς εξατομικευμένες επιλογές.

Κεφάλαιο 6

Συμπεράσματα και Μελλοντική Εργασία

6.1 Συμπεράσματα

Στα πλαίσια της παρούσας πτυχιακής εργασίας, αναπτύχθηκε η διαδικτυακή εφαρμογή *AI Financial Dashboard*, η οποία έχει σχεδιαστεί ώστε να παρέχει μια ολοκληρωμένη εμπειρία παρακολούθησης και ανάλυσης χρηματοοικονομικών δεδομένων με όσο το δυνατόν πιο απλοποιημένο τρόπο.

Στόχος της εφαρμογής είναι να διευκολύνει τόσο έμπειρους όσο και νέους χρήστες στον χώρο των χρηματοοικονομικών, δίνοντάς τους τη δυνατότητα να κατανοήσουν τάσεις στις αγορές, να παρακολουθούν κρυπτονομίσματα, μετοχές και δείκτες, και να κάνουν προβλέψεις με τη βοήθεια AI, χωρίς να χρειάζεται να αντιμετωπίζουν περιττή πολυπλοκότητα ή υπερφόρτωση πληροφοριών.

Η διεπαφή χρήστη έχει σχεδιαστεί με γνώμονα την απλότητα και την ευχρηστία. Μέσα από το Dashboard, ο χρήστης μπορεί να έχει άμεση εικόνα για την κατάσταση της αγοράς με διαδραστικά γραφήματα και ενημερωτικές κάρτες. Επιπλέον, οι λειτουργίες *favorites*, *alerts* και *daily news updates* παρέχουν τη δυνατότητα στον χρήστη να λαμβάνει έγκαιρα ενημερώσεις χωρίς να χρειάζεται να παραμένει συνεχώς μπροστά από την οθόνη, καθιστώντας την εμπειρία πιο άνετη και αποτελεσματική.

Συμπερασματικά, η εφαρμογή προσφέρει μια προσιτή και κατανοητή προσέγγιση στην ανάλυση χρηματοοικονομικών δεδομένων, εξισορροπώντας την ισχύ των εργαλείων AI με την απλότητα στην παρουσίαση των πληροφοριών. Με αυτόν τον τρόπο, ακόμη και νέοι χρήστες μπορούν να εμπλακούν ενεργά στις αγορές και να λαμβάνουν έγκυρες πληροφορίες χωρίς περιττό “θόρυβο” ή τεχνική πολυπλοκότητα.

6.2 Μελλοντική Επέκταση

Η εφαρμογή *AI Financial Dashboard* μπορεί να επεκταθεί περαιτέρω με στόχο την αύξηση της λειτουργικότητας και της προσβασιμότητας. Μελλοντικές δυνατότητες περιλαμβάνουν:

- Ενσωμάτωση εφαρμογής για κινητές συσκευές (mobile app), ώστε οι χρήστες να μπορούν

να παρακολουθούν τα αγαπημένα τους assets και τις ειδοποιήσεις ακόμα πιο εύκολα και εν κινήσει.

- Επέκταση των ειδοποιήσεων σε πραγματικό χρόνο (real-time notifications) για σημαντικές αλλαγές στις τιμές ή σημαντικά νέα γεγονότα.
- Ενσωμάτωση επιπλέον μοντέλων τεχνητής νοημοσύνης για πιο προχωρημένες προβλέψεις, ανάλυση κινδύνου και προτάσεις επένδυσης.
- Προσθήκη λειτουργιών εξατομίκευσης της διεπαφής (custom dashboards), ώστε κάθε χρήστης να μπορεί να δημιουργεί το δικό του περιβάλλον παρακολούθησης.
- Ανάπτυξη εργαλείων κοινωνικής δικτύωσης (social features), που θα επιτρέπουν στους χρήστες να μοιράζονται insights ή να συγκρίνουν τις επιλογές τους με άλλους επενδυτές.

Με αυτές τις βελτιώσεις, η εφαρμογή θα προσφέρει ακόμη πιο ολοκληρωμένη εμπειρία στον χρήστη, συνδυάζοντας ισχυρά εργαλεία ανάλυσης με ευχρηστία και ευελιξία.

Βιβλιογραφία

Γενικά για JavaScript και TypeScript

1. Cherny, B. *Programming TypeScript*. O'Reilly Media. Διαθέσιμο στο: <https://www.oreilly.com/library/view/programming-typescript/9781492037644/>
2. Bierman, G., Abadi, M., & Torgersen, M. (2014). *Understanding TypeScript*. Microsoft Research. Διαθέσιμο στο: https://dl.acm.org/doi/10.1007/978-3-662-44202-9_11
3. MDN Web Docs. *JavaScript Overview*. Mozilla Foundation. Διαθέσιμο στο: <https://developer.mozilla.org/en-US/docs/Web/JavaScript>

React και Frontend

1. React Team. *React Documentation*. Meta Open Source. Διαθέσιμο στο: <https://react.dev>
2. Vercel. *Next.js Documentation: React in Next.js*. Διαθέσιμο στο: <https://nextjs.org/docs/getting-started/react>
3. Vercel. *Next.js Documentation*. Διαθέσιμο στο: <https://nextjs.org/docs>
4. Tailwind. *Tailwing Documentation*. Διαθέσιμο στο : <https://v2.tailwindcss.com/docs>

MongoDB Atlas (Backend)

1. MongoDB Inc. *MongoDB Atlas Documentation*. Διαθέσιμο στο: <https://www.mongodb.com/docs/atlas/>

Inngest (Event-driven Flows)

1. Inngest Documentation. *Inngest Platform Overview*. Διαθέσιμο στο: <https://www.inngest.com/docs>

Δεδομένα και Market News

1. CoinGecko. *API Crypto Endpoint* Διαθέσιμο στο: <https://www.coingecko.com/>
2. Finnhub. *API Market Endpoint*. Διαθέσιμο στο: <https://finnhub.io/docs>
3. FunexClub Blog. *Bullish vs Bearish Market*. Διαθέσιμο στο: <https://blog.funexclub.com/difference-between-bearish-and-bullish-market/>