

ΣΥΓΧΡΟΝΑ ΛΕΙΤΟΥΡΓΙΚΑ ΣΥΣΤΗΜΑΤΑ • ΕΒΔΟΜΑΔΑ 1

Εισαγωγή & Βασική Αρχιτεκτονική Λ.Σ.

Ο θεμελιώδης ρόλος του Λειτουργικού Συστήματος στην Ανάπτυξη
Λογισμικού, Web & Video Games.

Τι είναι το Λειτουργικό Σύστημα (OS);

Η κεντρική γέφυρα επικοινωνίας μεταξύ υλικού (hardware) και εφαρμογών (software).



Διαχείριση Πόρων

Κατανόηση CPU, RAM, αποθήκευσης και περιφερειακών συσκευών.



Περιβάλλον Εκτέλεσης

Παροχή σταθερού περιβάλλοντος για browsers, IDEs και game engines.



Ασφάλεια & Έλεγχος

Δικαιώματα χρηστών, απομόνωση διεργασιών και προστασία δεδομένων.

Τι είναι το Λειτουργικό Σύστημα (Λ.Σ.);

- Σύνολο προγραμμάτων που διαχειρίζονται το υλικό (Hardware).
- Παρέχουν κοινές υπηρεσίες στις εφαρμογές (Software).
- **Η αναλογία:** Το Λ.Σ. είναι ο "διαχειριστής" μιας πολυκατοικίας (Hardware) για να ζουν αρμονικά οι ένοικοι (Εφαρμογές).
- Απαραίτητο για την επικοινωνία κώδικα/λογισμικού με τους πόρους του υπολογιστή (μέσω APIs & System Calls).

Αρχιτεκτονική Λ.Σ.: Πυρήνας & Κέλυφος

Πώς αλληλεπιδρούν τα επίπεδα του λειτουργικού συστήματος.

ΚΑΡΔΙΑ ΣΥΣΤΗΜΑΤΟΣ

Πυρήνας (Kernel)

Ο πυρήνας διαχειρίζεται απευθείας το υλικό, τους χρονισμούς της CPU και τις περιοχές μνήμης RAM, παραμένοντας αόρατος στον τελικό χρήστη.

ΔΙΕΠΑΦΗ

Κέλυφος (Shell / GUI)

Το περιβάλλον γραφικών (GUI) ή η κονσόλα εντολών (CLI) που επιτρέπει στον χρήστη και στις εφαρμογές να δίνουν εντολές στον πυρήνα.

Αρχιτεκτονική Στρωμάτων (Layered Architecture)

- **Hardware:** CPU, RAM, Δίσκοι, GPU, Κάρτα Δικτύου.
- **Πυρήνας (Kernel):** Αποκλειστική πρόσβαση στο hardware (Kernel Mode).
- **System Calls / Υπηρεσίες:** Μηχανισμοί επικοινωνίας.
- **Εφαρμογές (Software):** Browsers, IDEs, Game Engines (User Mode).
- **Σημαντικό:** Διαχωρισμός User Mode vs Kernel Mode για την ασφάλεια του συστήματος.

Βασικές Αρχιτεκτονικές Πυρήνα

Συγκριτική προσέγγιση μεταξύ Monolithic και Microkernel σχεδίασης.

Monolithic Kernel

Όλες οι υπηρεσίες του συστήματος (drivers, filesystem) τρέχουν στον ίδιο χώρο μνήμης.

□ Παράδειγμα: Linux, Windows NT. Υψηλή ταχύτητα εκτέλεσης.

Microkernel

Ελάχιστες βασικές λειτουργίες στον πυρήνα· οι υπόλοιπες τρέχουν ως ανεξάρτητες διεργασίες.

□ Παράδειγμα: Minix. Υψηλή σταθερότητα και ασφάλεια.

Η Καρδιά του Λ.Σ.: Ο Πυρήνας (Kernel)

- **Monolithic Kernel (Μονολιθικός):**
 - Όλες οι υπηρεσίες τρέχουν στον ίδιο χώρο μνήμης.
 - *Παράδειγμα:* Linux.
 - *Πλεονέκτημα:* Υψηλές ταχύτητες και επιδόσεις.
- **Microkernel (Μικροπυρήνας):**
 - Στον πυρήνα μόνο τα απολύτως απαραίτητα.
 - *Παράδειγμα:* Minix, QNX.
 - *Πλεονέκτημα:* Υψηλή σταθερότητα και ασφάλεια.

Σύνδεση με την Ειδικότητα

Γιατί ο Τεχνικός Εφαρμογών Πληροφορικής πρέπει να γνωρίζει Λ.Σ.;

Web Developers

Στήσιμο τοπικών servers, διαχείριση αρχείων και εργασία σε Linux/Unix περιβάλλοντα.

Multimedia & Video

Βελτιστοποίηση πόρων συστήματος (RAM/CPU) για rendering και επεξεργασία πολυμέσων.

Video Game Creators

Κατανόηση drivers καρτών γραφικών, διαχείρισης threads και cross-platform εκτέλεσης.

Σύνδεση Λ.Σ. με την Ειδικότητά μας

- **Web Developers:** Τοπική ανάπτυξη (Localhost), προσομοίωση servers, δικαιώματα αρχείων.
- **Multimedia Creators:** Βαριά διαχείριση πόρων (RAM, GPU), βελτιστοποίηση Multi-threading για rendering.
- **Game Developers:** Άμεση επαφή με APIs γραφικών (DirectX, Vulkan, Metal), διαχείριση χαμηλής καθυστέρησης (Low Latency).

Σύνοψη & Συμπεράσματα

- Το Λ.Σ. είναι ο θεμέλιος λίθος κάθε υπολογιστικού συστήματος.
- Η γνώση της αρχιτεκτονικής του βοηθάει στην αποδοτικότερη ανάπτυξη κώδικα.
- Ένας καλός developer κατανοεί τι συμβαίνει «κάτω από το καπό».

Ερωτήσεις & Συζήτηση

Είστε έτοιμοι για το εργαστηριακό σενάριο χαρτογράφησης υλικού και διεργασιών;

