



**ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΑΤΡΩΝ**

**ΤΜΗΜΑ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ ΚΑΙ  
ΤΕΧΝΟΛΟΓΙΑΣ ΥΠΟΛΟΓΙΣΤΩΝ**

**ΤΟΜΕΑΣ ΗΛΕΚΤΡΟΝΙΚΗΣ ΚΑΙ ΥΠΟΛΟΓΙΣΤΩΝ  
ΕΡΓΑΣΤΗΡΙΟ ΣΥΣΤΗΜΑΤΩΝ ΥΠΟΛΟΓΙΣΤΩΝ**

---

## **Αξιοποίηση υπηρεσιών ιδιωτικού νέφους στα πλαίσια του Διαδικτύου των Αντικειμένων**

Διπλωματική Εργασία

**Αθανάσιος Καλαθήρης του Μιχαήλ**

**Επιβλέπων: Κλεάνθης Θραμπουλίδης  
Πάτρα, Ιούλιος 2025**

---

Πανεπιστήμιο Πατρών, Τμήμα Ηλεκτρολόγων Μηχανικών και Τεχνολογίας Υπολογιστών.

Αθανάσιος Καλαθέρης του Μιχαήλ

© Ιούλιος 2025 – Με την επιφύλαξη παντός δικαιώματος.

Το σύνολο της εργασίας αποτελεί πρωτότυπο έργο, παραχθέν από τον Αθανάσιο Καλαθέρη του Μιχαήλ, και δεν παραβιάζει δικαιώματα τρίτων καθ' οιονδήποτε τρόπο. Αν η εργασία περιέχει υλικό, το οποίο δεν έχει παραχθεί από τον/την ίδιο/α, αυτό είναι ευδιάκριτο και αναφέρεται ρητώς εντός του κειμένου της εργασίας ως προϊόν εργασίας τρίτου, σημειώνοντας με παρομοίως σαφή τρόπο τα στοιχεία ταυτοποίησής του, ενώ παράλληλα βεβαιώνει πως στην περίπτωση χρήσης αυτούσιων γραφικών αναπαραστάσεων, εικόνων, γραφημάτων κ.λπ., έχει λάβει τη χωρίς περιορισμούς άδεια του κατόχου των πνευματικών δικαιωμάτων για την συμπερίληψη και επακόλουθη δημοσίευση του υλικού αυτού.

---

# Πιστοποίηση

Πιστοποιείται ότι η διπλωματική εργασία με τίτλο

## **Αξιοποίηση υπηρεσιών ιδιωτικού νέφους στα πλαίσια του Διαδικτύου των Αντικειμένων**

του φοιτητή του Τμήματος Ηλεκτρολόγων Μηχανικών και Τεχνολογίας  
Υπολογιστών

Αθάνασιου Καλαθέρη του Μιχαήλ

(Α.Μ.: 1019421)

παρουσιάστηκε δημόσια και εξετάστηκε στο Τμήμα Ηλεκτρολόγων Μηχανικών  
και Τεχνολογίας Υπολογιστών στις

\_\_\_/\_\_\_/\_\_\_

και εξετάστηκε από την ακόλουθη εξεταστική επιτροπή:

Κλεάνθης Θραμπουλίδης, Καθηγητής, Τμήμα Μηχανικών  
Υπολογιστών και Πληροφορικής

Χρήστος Φείδας, Καθηγητής, Τμήμα Ηλεκτρολόγων Μηχανικών  
και Τεχνολογίας Υπολογιστών

Ο Επιβλέπων

Ο Συνεπιβλέπων

Κλεάνθης Θραμπουλίδης  
Καθηγητής

Χρήστος Φείδας  
Καθηγητής

# Πρόλογος

## Ευχαριστίες

Αρχικά, εκφράζω τις θερμές μου ευχαριστίες στον επιβλέποντα καθηγητή της διπλωματικής μου, κ. Θραμπουλίδη Κλεάνθη, για την ευκαιρία που μου παρείχε να ασχοληθώ με το συγκεκριμένο αντικείμενο, καθώς και για την πολύτιμη καθοδήγηση και στήριξη που μου προσέφερε κατά τη διάρκεια εκπόνησης της εργασίας μου. Οι γνώσεις και οι εμπειρίες που αποκόμισα κατά τη διάρκεια αυτής της διαδικασίας αποτελούν πολύτιμο εφόδιο για τη μελλοντική μου πορεία.

Επιπλέον, ευχαριστώ θερμά τον συνάδελφο Τριανταφύλλου Αχιλλέα για την καθοδήγησή του και τη βοήθεια που μου παρείχε στην κατανόηση του συστήματος.

Τέλος, εκφράζω την ειλικρινή μου ευγνωμοσύνη στους γονείς μου, Μιχάλη και Ευαγγελία, καθώς και σε όλη την οικογένειά μου, για τη διαρκή πνευματική υποστήριξη και κατανόηση που έδειξαν καθ' όλη τη διάρκεια των σπουδών μου στο Πανεπιστήμιο Πατρών.

# Περίληψη

## Αξιοποίηση υπηρεσιών ιδιωτικού νέφους στα πλαίσια του Διαδικτύου των Αντικειμένων

**Ονοματεπώνυμο Φοιτητή**  
**Αθανάσιος Καλαθέρης**

**Ονοματεπώνυμο Επιβλέποντος**  
**Κλεάνθης Θραμπουλίδης**

Η παρούσα διπλωματική εργασία εξετάζει την αξιοποίηση υπηρεσιών ιδιωτικού cloud στο πλαίσιο του Διαδικτύου των Αντικειμένων (IoT). Ειδικότερα, η εργασία επικεντρώνεται στη χρήση του Kubernetes ως υποδομή ιδιωτικού cloud για την ανάπτυξη και διαχείριση μικροϋπηρεσιών (microservices) καθώς και πώς η ιδιωτική υποδομή αυτή θα μπορούσε να υλοποιηθεί σε ένα περιβαλλόν εργοστασίου. Στην συνέχεια σχεδιάστηκαν και υλοποιήθηκαν ένας αριθμός από μικροϋπηρεσίες (microservices) με στόχο τον έλεγχο και την παρακολούθηση μιας γραμμής παραγωγής, επιδεικνύοντας την αποτελεσματικότητα και την ευελιξία που παρέχουν οι τεχνολογίες cloud σε βιομηχανικές εφαρμογές.

Η καινοτομία του συστήματος έγκειται στη χρήση κυβερνοφυσικών microservices, οι οποίες αλληλεπιδρούν με ρομποτικούς βραχίονες μέσω HTTP requests. Κάθε microservice αντιπροσωπεύει ένα συγκεκριμένο βήμα παραγωγής, επιτυγχάνοντας ουσιαστικό διαχωρισμό (decoupling) ανάμεσα στη λογική των βημάτων και στους ρομποτικούς βραχίονες. Οι βραχίονες λειτουργούν ως απομακρυσμένα API endpoints που εκτελούν την κίνηση ή τη διεργασία που ζητούν τα microservices, χωρίς να χρειάζεται να γνωρίζουν οι ίδιοι τη σειρά ή τον τρόπο υλοποίησης των επιμέρους σταδίων. Επιπλέον, πραγματοποιήθηκε ενορχήστρωση των υπηρεσιών (service orchestration) για τον συγχρονισμό και το συντονισμό των διαφορετικών βημάτων παραγωγής, διασφαλίζοντας την ομαλή και ευέλικτη ροή της διαδικασίας.

# **Extensive English Summary**

## **Exploitation of Private Cloud Services in the IoT space**

**Student name, surname**

**Athanasios Kalatheris**

**Supervisor name, surname**

**Kleanthis Thramboulidis**

This thesis examines the utilization of private cloud services in the context of the Internet of Things (IoT). Specifically, the study focuses on the use of Kubernetes as a private cloud infrastructure for the development and management of microservices, as well as how this private infrastructure could be implemented within a factory environment. Subsequently, a number of microservices were designed and implemented with the aim of controlling and monitoring a production line, demonstrating the effectiveness and flexibility offered by cloud technologies in industrial applications.

The innovation of the system lies in the use of cyber-physical microservices, which interact with robotic arms via HTTP requests. Each microservice represents a specific production step, achieving substantial decoupling between the logic of the steps and the robotic arms. The robotic arms function as remote API endpoints that execute the movement or process requested by the microservices, without needing to know the sequence or the implementation details of the individual stages themselves. In addition, service orchestration was implemented to synchronize and coordinate the various production steps, ensuring a smooth and flexible workflow.

# Περιεχόμενα

<b>Ευρετήριο Εικόνων</b>	<b>ix</b>
<b>Ευρετήριο Πινάκων</b>	<b>xi</b>
<b>1 Εισαγωγή</b>	<b>1</b>
1.1 Αντικείμενο και στόχοι της διπλωματικής εργασίας . . . . .	1
1.2 Σχετικό Έργο . . . . .	1
1.3 Η μελέτη περίπτωσης . . . . .	2
1.4 Οργάνωση του κειμένου . . . . .	3
<b>2 Διαδίκτυο των Αντικειμένων και Κυβερνοφυσικά Συστήματα</b>	<b>5</b>
2.1 Εισαγωγή στο Διαδίκτυο των Αντικειμένων . . . . .	5
2.1.1 Η εξέλιξη του Web (από Web 1.0 έως Web 4.0) . . . . .	5
2.1.2 Βασικά επίπεδα (Layers) του IoT κατά Atzori . . . . .	6
2.1.3 Εφαρμογές και παραδείγματα . . . . .	6
2.2 Κυβερνοφυσικά Συστήματα (CPS) . . . . .	7
2.2.1 Βασικά Χαρακτηριστικά και Προκλήσεις . . . . .	7
2.2.2 Εφαρμογές και παραδείγματα CPS . . . . .	8
2.3 Σημαντικά ζητήματα και προκλήσεις . . . . .	8
2.3.1 Ασφάλεια και προστασία ιδιωτικότητας . . . . .	8
2.3.2 Διαλειτουργικότητα και πρότυπα . . . . .	8
2.3.3 Διαχείριση μεγάλου όγκου δεδομένων . . . . .	9
<b>3 Ιδιωτικές Υποδομές και Εικονοποίηση</b>	<b>11</b>
3.1 Εισαγωγή στην εικονοποίηση . . . . .	11
3.1.1 Σύνοψη ιστορική αναδρομή . . . . .	11
3.1.2 Βασικές τεχνικές . . . . .	11
3.2 Εικονικές Μηχανές (Virtual Machines) . . . . .	12
3.2.1 Δομή και λειτουργία . . . . .	12
3.2.2 Πλεονεκτήματα και μειονεκτήματα . . . . .	12
3.2.3 Χρήσεις και παραδείγματα . . . . .	13
3.3 Containers . . . . .	13
3.3.1 Βασικές αρχές και τεχνολογίες . . . . .	13
3.3.2 Πλεονεκτήματα και μειονεκτήματα . . . . .	13
3.3.3 Χρήσεις και παραδείγματα . . . . .	14
3.4 Σύγκριση Virtual Machines και Containers . . . . .	14
3.5 Προκλήσεις και τάσεις . . . . .	14
3.5.1 Ασφάλεια (Security) . . . . .	14
3.5.2 Διαχείριση κλίμακας (Scalability) . . . . .	15

<b>4</b>	<b>Kubernetes</b>	<b>17</b>
4.1	Βασικές αρχές του Kubernetes	17
4.2	Pods, Services και λοιποί πυρήνες μηχανισμοί	18
4.2.1	Αποθήκευση (Storage)	18
4.2.2	Ασφάλεια (Security)	18
4.3	Kubernetes και ενοποίηση με containers, VMs, IoT και CPS	18
4.3.1	Containers και Kubernetes	18
4.3.2	Virtual Machines και KubeVirt	19
4.3.3	IoT και Edge Computing	19
4.3.4	Cyber-Physical Systems (CPS)	19
4.4	Κλιμακούμενη ανάπτυξη και βέλτιστες πρακτικές	20
4.4.1	Autoscaling μηχανισμοί	20
4.4.2	Rolling Updates και Canary Deployments	20
4.4.3	Παρακολούθηση (Monitoring) και καταγραφή (Logging)	20
<b>5</b>	<b>Αξιοποίηση Κυβερνοφυσικών Υπηρεσιών</b>	<b>21</b>
5.1	Επισκόπηση Συστήματος	21
5.2	Στάδια Παραγωγής	21
5.3	Ρόλοι και Ευθύνες	22
5.3.1	Ρομπότ 1	22
5.3.2	Ρομπότ 2	22
5.3.3	Ρομπότ 3	23
5.3.4	Πάγκος Εργασίας 1	24
5.3.5	Πάγκος Εργασίας 2	25
5.4	Εργασίες Συναρμολόγησης	26
5.4.1	Εργασία Συναρμολόγησης 1	26
5.4.2	Εργασία Συναρμολόγησης 2	26
5.4.3	Εργασία Συναρμολόγησης 3	27
5.4.4	Εργασία Συναρμολόγησης 4	27
5.4.5	Εργασία Συναρμολόγησης 5	28
5.4.6	Εργασία Συναρμολόγησης 6	28
5.4.7	Εργασία Συναρμολόγησης 7	29
5.4.8	Εργασία Συναρμολόγησης 8	29
<b>6</b>	<b>Υλοποίηση των υποσυστημάτων με χρήση Kubernetes</b>	<b>31</b>
<b>7</b>	<b>Συμπεράσματα – Μελλοντική εργασία</b>	<b>33</b>



# Ευρετήριο Εικόνων

5.1 Διάγραμμα Ροής Ρομπότ 1 . . . . .	22
5.2 Διάγραμμα Ροής Ρομπότ 2 . . . . .	23
5.3 Διάγραμμα Ροής Ρομπότ 3 . . . . .	24
5.4 Διάγραμμα Ροής Πάγκου Εργασίας 1 . . . . .	24
5.5 Διάγραμμα Ροής Θέσεων Πάγκου Εργασίας 1 . . . . .	25
5.6 Διάγραμμα Ροής Πάγκου Εργασίας 2 . . . . .	25



# Ευρετήριο Πινάκων

3.1 Συνοπτική σύγκριση Virtual Machines και Containers . . . . .	14
--	----



# 1. Εισαγωγή

## 1.1 Αντικείμενο και στόχοι της διπλωματικής εργασίας

Αντικείμενο της παρούσας εργασίας είναι η μελέτη και η υλοποίηση μιας παραδοσιακής γραμμής συναρμολόγησης στη βιομηχανία, αξιοποιώντας τεχνολογίες του Διαδικτύου των Αντικειμένων και των Containers. Πιο αναλυτικά, εξετάστηκαν οι δυσκολίες και προκλήσεις που αντιμετωπίζει η βιομηχανία σε μια παραδοσιακή γραμμή συναρμολόγησης και πώς τεχνολογίες ιδιωτικών υποδομών μπορούν να υλοποιηθούν. Ιδιαίτερα, ο έλεγχος των φυσικών διεργασιών μέσω HTTP requests βασίστηκε σε υλοποίηση προηγούμενης διπλωματικής εργασίας, πάνω στην οποία έγιναν επεκτάσεις και βελτιώσεις για τις ανάγκες της παρούσας έρευνας. Ταυτόχρονα, καταλήξαμε στη χρήση του Apache Kafka για τη ροή μηνυμάτων μεταξύ των microservices, παράλληλα με τους Docker Containers και την πλατφόρμα ορχήστρωσης Kubernetes ως ιδιωτική υποδομή, με στόχο να διερευνηθούν οι δυνατότητες αξιοποίησής τους στον συγκεκριμένο τομέα.

Η διαδικασία αυτή υλοποιείται μέσα από την ανάπτυξη ενός κατανεμημένου συστήματος συναρμολόγησης, στο οποίο τα επιμέρους υποσυστήματα δεν είναι χωροταξικά συγκεντρωμένα και επικοινωνούν μεταξύ τους με πρωτόκολλα επιπέδου εφαρμογής και ανταλλαγή μηνυμάτων. Η επικοινωνία αυτή επιτυγχάνεται μέσω λογισμικού που αναπτύχθηκε και εκτελείται μέσα σε περιβάλλοντα Containers, τα οποία ορχηστρώνονται από το Kubernetes.

Το σύστημα συναρμολόγησης που χρησιμοποιήθηκε ως σενάριο μελέτης είναι το Gregor Office Chair Assembly System [1], το οποίο αποτελείται από τρεις ρομποτικούς βραχίονες και δύο τράπεζες εργασίας για τη συναρμολόγηση καρεκλών. Για την προσομοίωση του συστήματος, κατασκευάστηκε μια πειραματική διάταξη, ώστε να μελετηθούν οι δυνατότητες επικοινωνίας μεταξύ συσκευών με περιορισμένες υπολογιστικές δυνατότητες, χρησιμοποιώντας HTTP, Apache Kafka, Docker Containers και Kubernetes.

Μέσα από την εν λόγω μελέτη, εκτιμάται ότι η εφαρμογή των συγκεκριμένων τεχνολογιών μπορεί να προσφέρει ποικίλα πλεονεκτήματα στην ανάπτυξη λογισμικού για αντίστοιχα συστήματα στη βιομηχανία. Ενδεικτικά, η χρήση τους μπορεί να μειώσει το κόστος συναρμολόγησης, να επιταχύνει τον χρόνο ανάπτυξης του λογισμικού και να επιτρέψει την επαναχρησιμοποίηση ήδη υπάρχοντος κώδικα μέσω των Containers για τη δημιουργία νέων συστημάτων συναρμολόγησης. Τέλος, προέκυψαν διάφορα συμπεράσματα σχετικά με την εφαρμογή τους, τα οποία δείχνουν ότι η ενσωμάτωσή τους στη βιομηχανία μπορεί να αποφέρει σημαντικά οφέλη τόσο στη διαδικασία ανάπτυξης των συστημάτων όσο και στον τρόπο διαχείρισης αυτών από τους μηχανικούς που τα υποστηρίζουν.

## 1.2 Σχετικό Έργο

Στο πλαίσιο της παρούσας διπλωματικής εργασίας, ιδιαίτερα σημαντική υπήρξε η μελέτη και αξιοποίηση προηγούμενων σχετικών διπλωματικών εργασιών, οι οποίες εστίασαν σε τεχνολογίες Διαδικτύου των Αντικειμένων (IoT), containers και κυβερνοφυσικά συστήματα.

Η εργασία του φοιτητή Νικολάου Νικήτα με θέμα «Αξιοποίηση τεχνολογιών Διαδικτύου

των Αντικειμένων και containers σε κυβερνοφυσικά συστήματα» μελετά τις τεχνολογίες IoT και containers σε σχέση με την υλοποίηση μιας παραδοσιακής γραμμής συναρμολόγησης στη βιομηχανία. Εξετάζονται τα πρωτόκολλα CoAP και LwM2M, καθώς και η χρήση των Docker containers, με στόχο τη διερεύνηση της αξιοποίησής τους στον συγκεκριμένο τομέα. Σενάριο μελέτης αποτέλεσε το Gregor Office Chair Assembly System, με πειραματική διάταξη για τη μελέτη των δυνατοτήτων επικοινωνίας μεταξύ συσκευών με περιορισμένες υπολογιστικές δυνατότητες. Ο συγγραφέας καταλήγει στο ότι ο συγκεκριμένος σχεδιασμός συμβάλλει στη μείωση του κόστους συναρμολόγησης και του χρόνου ανάπτυξης λογισμικού, ενώ διευκολύνει την επαναχρησιμοποίηση λογισμικού μέσω των containers για νέα συστήματα συναρμολόγησης. Η συγκεκριμένη διπλωματική εργασία αποτέλεσε για εμένα το βασικό υπόβαθρο για την κατανόηση τόσο της περίπτωσης μελέτης Gregor, όσο και της αρχιτεκτονικής των microservices. Επιπλέον, βασίστηκα στις υλοποιήσεις της για να κατανοήσω τον τρόπο με τον οποίο πραγματοποιείται ο συγχρονισμός μεταξύ των επιμέρους υπηρεσιών.

Η εργασία του φοιτητή Δημητρίου Σπυριδών με τίτλο «Αξιοποίηση Cyber-Physical microservices σε συστήματα IoT» επικεντρώνεται στη μελέτη τεχνολογιών IoT και στην υπηρεσιοκεντρική προσέγγιση για την αξιοποίησή τους σε συστήματα συναρμολόγησης. Στο πλαίσιο αυτής της εργασίας υλοποιήθηκαν πρωτογενή Cyber-Physical Microservices που ελέγχουν τα μέρη του συστήματος συναρμολόγησης, καθώς και σύνθετα Cyber-Physical Microservices που αξιοποιούν τα πρωτογενή για την παροχή πιο σύνθετων λειτουργιών. Οι υπηρεσίες αυτές εξάγονται στο δίκτυο μέσω HTTP endpoints, τα οποία αποτελούν το REST API της υποδομής. Μέσω αυτού του API, είναι δυνατή τόσο η ενορχήστρωση των υπηρεσιών (service orchestration) όσο και η ανάπτυξη χορογραφίας υπηρεσιών (service choreography) για την κατασκευή ενός συστήματος συναρμολόγησης. Από τη συγκεκριμένη διπλωματική εργασία χρησιμοποίησα τα HTTP endpoints, πάνω στην οποία βασίστηκα αλλά και βελτίωσα για τις ανάγκες της ενορχήστρωσης των υπηρεσιών (service orchestration). Συνοψίζοντας, τα συμπεράσματα και οι τεχνικές που αντλήθηκαν από τα παραπάνω έργα αποτέλεσαν θεμέλιο λίθο για τη σχεδίαση και υλοποίηση του παρόντος συστήματος, το οποίο ενσωματώνει τις πλέον σύγχρονες πρακτικές στον χώρο των κυβερνοφυσικών συστημάτων και της βιομηχανικής παραγωγής.

### 1.3 Η μελέτη περίπτωσης

Ως μελέτη περίπτωσης επιλέχθηκε το παράδειγμα ελέγχου φυσικών μερών για τη συναρμολόγηση μιας καρέκλας γραφείου, όπως απεικονίζεται στο Σχήμα 1.1. Στο σενάριο αυτό, ρομποτικοί βραχίονες και τράπεζες εργασίας συνεργάζονται ώστε να εκτελέσουν τις απαραίτητες εντολές για τη συναρμολόγηση του τελικού προϊόντος. Τα βασικά εξαρτήματα που απαιτούνται για τη συναρμολόγηση φτάνουν είτε μέσω ιμάντων μεταφοράς — στην περίπτωση των μεγαλύτερων εξαρτημάτων όπως το μαξιλάρι — είτε είναι ήδη τοποθετημένα σε θέσεις δίπλα στους ρομποτικούς βραχίονες για τα μικρότερα μέρη, όπως οι ρόδες.

Η κατανομή των εργασιών έχει σχεδιαστεί έτσι ώστε να αξιοποιούνται με τον βέλτιστο δυνατό τρόπο οι δυνατότητες των ρομποτικών βραχιόνων στη διαδικασία συναρμολόγησης μιας καρέκλας. Στο δοκιμαστικό περιβάλλον, για τρεις παραγγελίες καρεκλών, κάθε ρομποτικός βραχίονας αναλαμβάνει συγκεκριμένο στάδιο της συναρμολόγησης, επιτυγχάνοντας παράλληλη εκτέλεση των εργασιών και αυξάνοντας την αποδοτικότητα της γραμμής παραγωγής.

Η διαδικασία ξεκινά με το κάτω μέρος της καρέκλας, το οποίο αποτελείται από τη βάση με τα πόδια και τις ρόδες. Ο ρομποτικός βραχίονας 1 παραλαμβάνει τη βάση των ποδιών και την τοποθετεί στην πρώτη μέγγενη, όπου προσαρμόζονται τα πέντε πόδια ένα προς ένα, με τη μέγγενη να περιστρέφεται ώστε να επιτρέπει τη σταδιακή τοποθέτηση. Η περιστρεφόμενη τριγωνική τράπεζα εργασίας 1 στη συνέχεια μεταφέρει το ημιτελές προϊόν στον ρομποτικό βραχίονα 2, ο οποίος τοποθετεί τις ρόδες και το αμορτισέρ. Ακολουθεί το στάδιο τοποθέτησης του καθίσματος.

Καθώς ο ρομποτικός βραχίονας 2 ολοκληρώνει το έργο του, ο ρομποτικός βραχίονας 1

παράλληλα επεξεργάζεται το κάθισμα στη δεύτερη τράπεζα εργασίας. Μόλις είναι έτοιμο, το κάθισμα μεταφέρεται και τοποθετείται πάνω στο αμορτισέρ του ημιτελούς προϊόντος από τον ρομποτικό βραχίονα 2. Έπειτα, η τράπεζα εργασίας 1 περιστρέφεται ξανά, οδηγώντας το προϊόν μπροστά στον ρομποτικό βραχίονα 3, ο οποίος ολοκληρώνει τη διαδικασία συναρμολόγησης τοποθετώντας την πλάτη και τα μπράτσα της καρέκλας.

Είναι σημαντικό να σημειωθεί ότι μόλις ένας ρομποτικός βραχίονας ολοκληρώσει το έργο του για μία καρέκλα και εφόσον υπάρχει επόμενη παραγγελία, ξεκινά άμεσα τη διαδικασία για το επόμενο προϊόν. Σε περιπτώσεις που δεν υπάρχουν εξαρτήσεις, οι βραχίονες μπορούν να ξεκινούν άμεσα την εργασία τους, επιτρέποντας ακόμα μεγαλύτερη παράλληλη εκτέλεση. Η ταυτόχρονη συναρμολόγηση έως και τριών καρεκλών και η χρήση δύο τραπεζών εργασίας αποτελούν ενδεικτικά παραδείγματα αύξησης της αποδοτικότητας και της ευελιξίας της γραμμής παραγωγής.

## 1.4 Οργάνωση του κειμένου

Η δομή της παρούσας διπλωματικής εργασίας έχει ως εξής:

- **Κεφάλαιο 1: Εισαγωγή** Παρουσιάζεται το αντικείμενο, οι στόχοι και το γενικό πλαίσιο της εργασίας, καθώς και η μεθοδολογική προσέγγιση που ακολουθήθηκε.
- **Κεφάλαιο 2: Κυβερνοφυσικά Συστήματα και Διαδίκτυο των Αντικειμένων (IoT)** Περιλαμβάνει ιστορική αναδρομή και θεωρητικές πληροφορίες σχετικά με τα κυβερνοφυσικά συστήματα και το IoT, αναδεικνύοντας τις βασικές αρχές, τις τεχνολογικές εξελίξεις και τις προκλήσεις στον τομέα.
- **Κεφάλαιο 3: Ιδιωτικές Υποδομές και Εικονοποίηση (Virtualization)** Παρουσιάζονται βασικές έννοιες της εικονοποίησης και των ιδιωτικών υποδομών, με έμφαση σε τεχνολογίες όπως το OpenStack και το Kubernetes. Αναλύεται η διαδικασία επιλογής του Kubernetes ως ιδιωτική υποδομή και περιγράφεται η υλοποίηση του σε περιβάλλον εργοστασίου.
- **Κεφάλαιο 4: Αξιοποίηση Κυβερνοφυσικών Υπηρεσιών** Αναφέρεται στη μελέτη περίπτωσης της γραμμής παραγωγής της καρέκλας Γρεγορ, παρουσιάζοντας τα επιμέρους βήματα και τη λειτουργία του συστήματος με αναλυτικό τρόπο καθώς και ο τρόπος με τον οποίο θα αξιοποιηθούν.
- **Κεφάλαιο 5: Υλοποίηση Ενορχήστρωσης και Αξιοποίηση της Ιδιωτικής Υποδομής** Περιγράφεται η διαδικασία ενορχήστρωσης, δίνοντας έμφαση στις τεχνικές υλοποίησης καθώς και την αξιοποίηση της ιδιωτικής υποδομής για τις μικρουπηρεσίες.
- **Κεφάλαιο 6: Συμπεράσματα και Μελλοντικές Επεκτάσεις** Παρουσιάζονται τα βασικά συμπεράσματα της εργασίας και προτείνονται ενδεχόμενες μελλοντικές κατευθύνσεις και προεκτάσεις για περαιτέρω έρευνα.





## **2. Διαδίκτυο των Αντικειμένων και Κυβερνοφυσικά Συστήματα**

Το Διαδίκτυο των Αντικειμένων (Internet of Things ή IoT) αποτελεί έναν από τους σημαντικότερους παράγοντες εξέλιξης της σύγχρονης τεχνολογίας και πληροφορικής. Παράλληλα, η έννοια των Κυβερνοφυσικών Συστημάτων (Cyber-Physical Systems ή CPS) έχει έρθει στο προσκήνιο τα τελευταία χρόνια, καθώς συνδυάζει συσκευές του φυσικού κόσμου με την υπολογιστική ισχύ και τις δυνατότητες δικτύωσης που χαρακτηρίζουν την ψηφιακή εποχή. Σε αυτό το κεφάλαιο παρουσιάζονται οι βασικές έννοιες, οι αρχές λειτουργίας και τα πλεονεκτήματα που προκύπτουν από την ενσωμάτωση τόσο του IoT όσο και των CPS σε διάφορους τομείς.

### **2.1 Εισαγωγή στο Διαδίκτυο των Αντικειμένων**

Το Internet of Things (IoT) αναφέρεται σε ένα οικοσύστημα όπου φυσικές συσκευές, αισθητήρες και άλλα αντικείμενα συνδέονται και ανταλλάσσουν δεδομένα μέσω του Διαδικτύου [2]. Η ιδέα αυτή θέτει τις βάσεις για την έξυπνη διασύνδεση και αλληλεπίδραση μεταξύ ποικίλων συστημάτων, από βιομηχανικές εφαρμογές έως έξυπνα σπίτια, προκειμένου να δημιουργηθούν νέες υπηρεσίες και βελτιστοποιημένες διαδικασίες [3].

Σύμφωνα με εκτιμήσεις, ο αριθμός των συνδεδεμένων συσκευών στο Διαδίκτυο αυξήθηκε ραγδαία την τελευταία δεκαετία. Ειδικότερα, υπολογίζεται πως το 2010 υπήρχαν περίπου 12.5 δισεκατομμύρια συνδεδεμένες συσκευές παγκοσμίως, ενώ προβλεπόταν να φτάσουν τα 50 δισεκατομμύρια έως το 2020 [4]. Αυτή η τάση αναμένεται να συνεχιστεί, καθώς τεχνολογίες όπως το Cloud Computing και η Artificial Intelligence γίνονται όλο και πιο προσιτές, συμβάλλοντας στην ευρύτερη υιοθέτηση του IoT σε διάφορους τομείς.

#### **2.1.1 Η εξέλιξη του Web (από Web 1.0 έως Web 4.0)**

Ιστορικά, η εξέλιξη του Παγκόσμιου Ιστού (World Wide Web) χωρίζεται σε διάφορες φάσεις: Web 1.0, Web 2.0, Web 3.0 και Web 4.0. Στο Web 1.0, το περιεχόμενο ήταν κυρίως στατικό, επιτρέποντας μονόδρομη αλληλεπίδραση (παροχή πληροφορίας από διακομιστές σε χρήστες). Η μετάβαση στο Web 2.0 εστίασε στην αλληλεπίδραση και τη δημιουργία περιεχομένου από τους ίδιους τους χρήστες, οδηγώντας στην ανάπτυξη κοινωνικών δικτύων και πλατφορμών ανταλλαγής περιεχομένου.

Η επόμενη γενιά, το Web 3.0, χαρακτηρίζεται από τη «σημασιολογική» διάσταση, όπου οι μηχανές και οι εφαρμογές μπορούν να «κατανοήσουν» καλύτερα τα δεδομένα και τις έννοιες, επιτρέποντας πιο ευφυείς υπηρεσίες. Τέλος, το Web 4.0 περιγράφεται συχνά ως «υπεύθυνες» πλατφόρμες, οι οποίες συνδέουν ανθρώπους και συσκευές σε πραγματικό χρόνο και ανταλλάσσουν πολυδιάστατα δεδομένα, γεγονός που συνδέεται άμεσα με την ανάπτυξη του IoT, της Cloud Computing τεχνολογίας και προηγμένων αλγορίθμων Artificial Intelligence.

### 2.1.2 Βασικά επίπεδα (Layers) του IoT κατά Atzori

Η αρχιτεκτονική του IoT μπορεί να περιγραφεί μέσα από πολλά μοντέλα (layers), προκειμένου να γίνει κατανοητή η ροή και η επεξεργασία των δεδομένων. Σύμφωνα με τους Atzori et al. [5], το IoT μπορεί να αναλυθεί σε πέντε επίπεδα, ως εξής:

1. **Επίπεδο Αντικειμένων (Objects):** Περιλαμβάνει όλους τους αισθητήρες, ενεργοποιητές και γενικότερα τις φυσικές IoT συσκευές που συλλέγουν δεδομένα από το περιβάλλον ή επιδρούν σε αυτό.
2. **Επίπεδο Αφαίρεσης Αντικειμένων (Object Abstraction):** Αφορά τις τεχνικές και τους μηχανισμούς που επιτρέπουν την αφαίρεση (abstraction) των δεδομένων, ώστε να μπορούν να διακινηθούν και να υποστούν επεξεργασία ανεξάρτητα από το φυσικό αντικείμενο ή τα επιμέρους πρωτόκολλα επικοινωνίας.
3. **Επίπεδο Διαχείρισης Υπηρεσιών (Service Management):** Περιλαμβάνει τις λειτουργίες που σχετίζονται με την προσφορά, ανακάλυψη, διαχείριση και εκτέλεση υπηρεσιών πάνω στα δεδομένα που προέρχονται από τα αντικείμενα.
4. **Επίπεδο Σύνθεσης Υπηρεσιών (Service Composition):** Σε αυτό το επίπεδο, οι επιμέρους υπηρεσίες συνδυάζονται ή «συντίθενται» για να δημιουργήσουν πιο πολύπλοκες και ολοκληρωμένες εφαρμογές IoT, προσφέροντας προηγμένες λειτουργίες στους χρήστες.
5. **Επίπεδο Εφαρμογών (Applications):** Αφορά το «τελικό» στάδιο, όπου οι εφαρμογές (smart home, smart city, industrial IoT κ.λπ.) αξιοποιούν τα δεδομένα και τις υπηρεσίες, παρέχοντας ουσιαστικές λύσεις και λειτουργικότητα στους τελικούς χρήστες.

Άλλοι ερευνητές προτείνουν ένα πιο συνοπτικό μοντέλο τριών επιπέδων (Perception, Network, Application) [2], ενώ ορισμένες εταιρείες, όπως η Cisco, περιγράφουν αναλυτικά μοντέλα πέντε ή έξι επιπέδων με έμφαση στη διαχείριση δικτύων edge, στον διαχωρισμό των πυλών (gateways) και στη σημασιολογική ανάλυση [6]. Σε κάθε περίπτωση, η ασφάλεια και η διαχείριση μεγάλων ποσοτήτων δεδομένων (Big Data) αποτελούν κοινές προκλήσεις που διατρέχουν όλα τα επίπεδα του IoT. Καθώς ο αριθμός των συνδεδεμένων συσκευών συνεχίζει να αυξάνεται, η ανάγκη για αξιόπιστα, ευέλικτα και ασφαλή συστήματα θα γίνεται ολοένα και πιο επιτακτική.

### 2.1.3 Εφαρμογές και παραδείγματα

Οι εφαρμογές του IoT καλύπτουν ένα ευρύ φάσμα:

- **Έξυπνα σπίτια:** Η αυτόματη ρύθμιση φωτισμού, θέρμανσης/κλιματισμού και η διαχείριση οικιακών συσκευών επιτρέπουν βελτιωμένη άνεση και εξοικονόμηση ενέργειας.
- **Έξυπνες πόλεις:** Αισθητήρες ενσωματωμένοι στο αστικό περιβάλλον συλλέγουν δεδομένα για την κυκλοφορία, τη ρύπανση ή τη διαχείριση απορριμμάτων, βελτιώνοντας τις παρεχόμενες υπηρεσίες και την ποιότητα ζωής.
- **Βιομηχανική παραγωγή:** Σε βιομηχανικά περιβάλλοντα, το IoT αξιοποιείται για την παρακολούθηση μηχανημάτων, την προληπτική συντήρηση (predictive maintenance) και τη βελτίωση της αποδοτικότητας των γραμμών παραγωγής.
- **Υγεία:** Φορετές συσκευές (wearables) και αισθητήρες παρακολουθούν ζωτικές ενδείξεις ασθενών και ενημερώνουν σε πραγματικό χρόνο ιατρικό προσωπικό για πιθανές επιπλοκές.

## 2.2 Κυβερνοφυσικά Συστήματα (CPS)

Τα κυβερνοφυσικά συστήματα (CPS) αποτελούν μία ολοένα και πιο σημαντική κατηγορία συστημάτων, τα οποία ενοποιούν αλληλεπιδράσεις μεταξύ του φυσικού και του ψηφιακού κόσμου [7], [8], [9]. Σε γενικές γραμμές, ένα CPS περιλαμβάνει υπολογιστικούς πόρους (π.χ. ενσωματωμένα συστήματα, δικτυακές υποδομές) και φυσικές διεργασίες (π.χ. βιομηχανικός εξοπλισμός, αισθητήρες, ρομπότ), τα οποία επικοινωνούν και λειτουργούν με στενή σύζευξη. Στόχος αυτής της ενσωμάτωσης είναι η επίτευξη υψηλών επιπέδων ευφυΐας, αυτονομίας και πραγματικού χρόνου ανταπόκρισης σε σύνθετα περιβάλλοντα.

Σύμφωνα με την ανάλυση του Lee [7], τα κυβερνοφυσικά συστήματα συναντώνται σε ένα ευρύ φάσμα εφαρμογών, όπως η βιομηχανική αυτοματοποίηση, τα συστήματα μεταφορών, τα ενεργειακά δίκτυα (smart grid), καθώς και η ιατρική παρακολούθηση και υποστήριξη. Βασικό χαρακτηριστικό είναι η συνεχής ανατροφοδότηση (feedback) μεταξύ φυσικών και ψηφιακών οντοτήτων, έτσι ώστε η απόκριση του συστήματος να είναι δυναμική και εξαρτώμενη από τις εκάστοτε συνθήκες.

Γνωρίζουμε πως τα CPS αποτελούν την επόμενη «υπολογιστική επανάσταση», [8][8] αφού οι απαιτήσεις σε θέματα συγχρονισμού, αξιοπιστίας και ασφάλειας είναι ιδιαίτερες αυξημένες συγκριτικά με τα παραδοσιακά IT ή ακόμα και τα IoT συστήματα. Ειδικότερα, η στενή σύζευξη μεταξύ λογισμικού και φυσικών διεργασιών επιβάλλει την ανάπτυξη νέων μεθοδολογιών σχεδίασης, οι οποίες λαμβάνουν υπόψη περιορισμούς όπως ο πραγματικός χρόνος (real time), η ενεργειακή αποδοτικότητα, αλλά και η προβλεψιμότητα της συμπεριφοράς του συστήματος.

Παράλληλα, η ολοένα αυξανόμενη δημοτικότητα των CPS οφείλεται και στις τεχνολογικές εξελίξεις στους τομείς του cloud computing [9], της ανάλυσης μεγάλων δεδομένων (big data), καθώς και των ασύρματων επικοινωνιών. Με αυτόν τον τρόπο, τα κυβερνοφυσικά συστήματα μπορούν να υλοποιήσουν πολύπλοκες λειτουργίες, όπως η προληπτική συντήρηση (predictive maintenance) σε βιομηχανικά περιβάλλοντα.

### 2.2.1 Βασικά Χαρακτηριστικά και Προκλήσεις

- **Πραγματικός χρόνος (real-time):** Η αλληλεπίδραση με τον φυσικό κόσμο απαιτεί χρονικά περιορισμένες αποκρίσεις, ώστε το σύστημα να αντιδρά με ακρίβεια σε μεταβαλλόμενες συνθήκες.
- **Ασφάλεια (security):** Καθώς τα CPS συνδέονται ολοένα και περισσότερο σε δίκτυα, η κυβερνοασφάλεια αποτελεί κρίσιμο ζήτημα. Οι επιθέσεις μπορούν να έχουν σοβαρές επιπτώσεις στον φυσικό κόσμο, όπως διακοπές σε βιομηχανικά συστήματα ή συστήματα υγειονομικής περίθαλψης.
- **Αξιοπιστία (reliability):** Η αδιάλειπτη λειτουργία του συστήματος είναι ουσιώδης, ιδίως σε εφαρμογές όπου η δυσλειτουργία ενέχει σημαντικούς κινδύνους (π.χ. στα έξυπνα δίκτυα ενέργειας ή στην αυτόνομη οδήγηση).
- **Επεκτασιμότητα (scalability):** Τα CPS καλούνται να διαχειριστούν μεγάλους όγκους δεδομένων και μεγάλο αριθμό συσκευών, διατηρώντας παράλληλα υψηλά επίπεδα απόδοσης.

Λαμβάνοντας υπόψη τις ανωτέρω παραμέτρους, τα κυβερνοφυσικά συστήματα αναμένεται να αποτελέσουν την επόμενη γενιά εφαρμογών σε πληθώρα τομέων, συνδυάζοντας ισχυρούς υπολογιστικούς πόρους, συνεχείς ροές δεδομένων και έξυπνα φυσικά στοιχεία. Η έρευνα επικεντρώνεται, μεταξύ άλλων, σε μεθόδους ολοκλήρωσης (integration) μεταξύ λογισμικού και υλισμικού, σε τεχνικές ανθεκτικότητας (fault tolerance) και σε προσεγγίσεις που εξασφαλίζουν την ασφάλεια και την αξιοπιστία σε όλες τις φάσεις λειτουργίας του συστήματος.

Τα Κυβερνοφυσικά Συστήματα (Cyber-Physical Systems) αποτελούν έναν ευρύτερο όρο που περιγράφει την ενοποίηση της υπολογιστικής ισχύος (κυβερνοχώρος) με φυσικές διεργασίες. Σε ένα CPS, τα φυσικά αντικείμενα και οι λειτουργίες τους ελέγχονται στενά από αλγορίθμους που εκτελούνται σε υπολογιστικά συστήματα, με στόχο την υλοποίηση ενός ολοκληρωμένου συστήματος ελέγχου και αυτοματισμού.

### 2.2.2 Εφαρμογές και παραδείγματα CPS

- **Έξυπνα εργοστάσια:** Ρομποτικά συστήματα και αισθητήρες ελέγχουν κάθε στάδιο παραγωγής, ενώ οι αποφάσεις για τη γραμμή παραγωγής λαμβάνονται δυναμικά, με βάση την κατάσταση των μηχανημάτων και την παραγωγική ζήτηση.
- **Αυτόνομα οχήματα:** Ένας συνδυασμός αισθητήρων (π.χ. κάμερες, LiDAR, RADAR) και αλγορίθμων τεχνητής νοημοσύνης (AI) επιτρέπει στα οχήματα να αλληλεπιδρούν με το περιβάλλον και να λαμβάνουν αποφάσεις σε πραγματικό χρόνο. Το σύστημα ελέγχου (κυβερνοχώρος) συνδέεται διαρκώς με τις φυσικές κινήσεις του οχήματος.
- **Έξυπνα δίκτυα ενέργειας:** Τα δίκτυα ηλεκτρικής ενέργειας γίνονται ολοένα πιο «έξυπνα», χάρη σε αισθητήρες και συστήματα ελέγχου που παρακολουθούν σε πραγματικό χρόνο την κατανάλωση ενέργειας. Οι ανανεώσιμες πηγές ενέργειας ενσωματώνονται δυναμικά, ενώ κεντρικά συστήματα και αλγόριθμοι βελτιστοποίησης ρυθμίζουν τη λειτουργία του δικτύου.
- **Υγειονομικά συστήματα:** Συνδυάζοντας φορητές συσκευές παρακολούθησης (π.χ. wearable sensors), βάσεις ιατρικών δεδομένων και ρομποτικές συσκευές, επιτυγχάνεται μια συνεχής αλληλεπίδραση μεταξύ θεραπευτικών πρακτικών και πραγματικού χρόνου παρακολούθησης της κατάστασης του ασθενή.

## 2.3 Σημαντικά ζητήματα και προκλήσεις

### 2.3.1 Ασφάλεια και προστασία ιδιωτικότητας

Τόσο τα συστήματα IoT όσο και τα CPS εκτίθενται σε πλήθος κινδύνων ασφαλείας. Ο μεγάλος αριθμός διασυνδεδεμένων συσκευών αυξάνει την πιθανότητα επίθεσης (attack surface), ενώ ενδεχόμενη πρόσβαση σε ευαίσθητα δεδομένα μπορεί να έχει σημαντικές επιπτώσεις στην ιδιωτικότητα των χρηστών. Επιπλέον, σε ένα CPS, επιθέσεις στον κυβερνοχώρο μπορούν να προκαλέσουν φυσικές ζημιές, καθώς η ψηφιακή διάσταση ελέγχει τη φυσική.

Για αυτόν τον λόγο, απαιτούνται πρωτόκολλα κρυπτογράφησης, συστήματα ελέγχου πρόσβασης και μηχανισμοί συνεχούς παρακολούθησης της δικτυακής κίνησης. Η πρόκληση έγκειται στη διατήρηση της ισορροπίας ανάμεσα στην υψηλή ασφάλεια και την ομαλή λειτουργία/επέκταση του δικτύου.

### 2.3.2 Διαλειτουργικότητα και πρότυπα

Η ραγδαία αύξηση των συσκευών και η ύπαρξη πολλών κατασκευαστών δημιουργούν την ανάγκη για πρότυπα επικοινωνίας, έτσι ώστε ετερογενή συστήματα να μπορούν να συνεργάζονται. Ορισμένα γνωστά πρωτόκολλα IoT περιλαμβάνουν το MQTT (Message Queuing Telemetry Transport), το CoAP (Constrained Application Protocol) και το HTTP (Hypertext Transfer Protocol).

Ωστόσο, σε περιπτώσεις CPS, οι απαιτήσεις για χρονική ακρίβεια και αξιοπιστία είναι ακόμη πιο αυστηρές, οδηγώντας σε ανάγκη για εξειδικευμένα πρωτόκολλα πραγματικού χρόνου (real-time protocols). Παράλληλα, η ευρεία αποδοχή συγκεκριμένων προτύπων θα επιταχύνει την ανάπτυξη λύσεων και θα διευκολύνει την ενσωμάτωση νέων τεχνολογιών.

### **2.3.3 Διαχείριση μεγάλου όγκου δεδομένων**

Καθώς οι αισθητήρες και οι συσκευές IoT/CPS παράγουν τεράστιο όγκο δεδομένων, η επεξεργασία και η αποθήκευση τους αποτελεί πρόκληση. Οι παραδοσιακές βάσεις δεδομένων συχνά δεν επαρκούν, οπότε η αξιοποίηση τεχνολογιών Big Data, cloud computing και edge/fog computing θεωρείται απαραίτητη. Ειδικά στα CPS, η έγκαιρη επεξεργασία των δεδομένων είναι κρίσιμη, επειδή οι αποφάσεις επηρεάζουν το φυσικό περιβάλλον σε πραγματικό χρόνο και τυχόν καθυστέρηση μπορεί να επιφέρει σφάλματα ή κινδύνους για την ασφάλεια.



## 3. Ιδιωτικές Υποδομές και Εικονοποίηση

Η εικονοποίηση (virtualization) αποτελεί μια θεμελιώδη τεχνολογία στον χώρο της πληροφορικής, επιτρέποντας τη δημιουργία πολλαπλών περιβαλλόντων (λειτουργικών συστημάτων ή εφαρμογών) πάνω σε μια κοινή φυσική υποδομή. Με αυτόν τον τρόπο, αξιοποιείται καλύτερα η διαθέσιμη υπολογιστική ισχύς, ενώ ταυτόχρονα παρέχονται δυνατότητες απομόνωσης, φορητότητας και ευελιξίας. Τα τελευταία χρόνια, ιδιαίτερη έμφαση δίνεται στις Εικονικές Μηχανές (Virtual Machines) και στα containers, δύο εναλλακτικές προσεγγίσεις εικονοποίησης που επιλύουν διαφορετικά προβλήματα και εξυπηρετούν διαφορετικές ανάγκες. Σε αυτό το κεφάλαιο, θα εξετάσουμε τις βασικές αρχές της εικονοποίησης, θα επικεντρωθούμε στα πλεονεκτήματα και τις προκλήσεις που σχετίζονται με τις virtual machines και τα containers και θα παρουσιάσουμε χαρακτηριστικές χρήσεις τους σε σύγχρονα περιβάλλοντα υπολογιστών και υποδομών cloud.

### 3.1 Εισαγωγή στην εικονοποίηση

Η εικονοποίηση (virtualization) είναι η διαδικασία διαχωρισμού των φυσικών πόρων (π.χ. CPU, μνήμη, αποθηκευτικός χώρος, δικτυακές διεπαφές) από τις λογικές λειτουργικές οντότητες που τους χρησιμοποιούν. Μέσα από ένα ενδιάμεσο επίπεδο λογισμικού, που συχνά ονομάζεται hypervisor, δημιουργούνται αυτόνομα, εικονικά περιβάλλοντα, τα οποία «νομίζουν» ότι έχουν πρόσβαση σε όλη την υποδομή. Ωστόσο, στην πραγματικότητα, το σύστημα ελέγχει και κατανέμει τους πόρους ανάλογα με τις ανάγκες και τις πολιτικές διαχείρισης.

#### 3.1.1 Σύντομη ιστορική αναδρομή

Η πρώτη μορφή εικονοποίησης εμφανίστηκε στη δεκαετία του 1960 σε υπολογιστές mainframe της IBM, όπου οι χρήστες μοιράζονταν υπολογιστική ισχύ μέσω ξεχωριστών εικονικών περιβαλλόντων. Με την πάροδο του χρόνου και τη ραγδαία αύξηση της υπολογιστικής ικανότητας των x86 αρχιτεκτονικών, η εικονοποίηση βγήκε από το πλαίσιο των κεντρικών υπολογιστών (mainframes) και έγινε προσιτή στα servers ευρείας χρήσης. Από τότε, έχουν αναπτυχθεί διαφορετικά μοντέλα και τεχνολογίες εικονοποίησης, με πιο δημοφιλείς υλοποιήσεις την εικονοποίηση επιπέδου λειτουργικού συστήματος (OS-level virtualization), τις virtual machines και, πιο πρόσφατα, τα containers.

#### 3.1.2 Βασικές τεχνικές

- **Εικονοποίηση υλικού (hardware-level virtualization):** Χρησιμοποιεί έναν hypervisor (π.χ. VMware ESXi, KVM, Hyper-V) ο οποίος αναλαμβάνει τη δρομολόγηση των εντολών μεταξύ του φυσικού επεξεργαστή και των εικονικών μηχανών.
- **Εικονοποίηση επιπέδου λειτουργικού συστήματος (OS-level virtualization):** Χρησιμοποιεί τις δυνατότητες του πυρήνα του λειτουργικού συστήματος για να δημιουργήσει πολλαπλά λογικά περιβάλλοντα, όπως συμβαίνει με τα containers.

- **Παραλλαγές paravirtualization:** Εδώ το λειτουργικό σύστημα μέσα στην εικονική μηχανή (guest OS) είναι τροποποιημένο, ώστε να υποστηρίζει αποτελεσματικότερα τις λειτουργίες του hypervisor, όπως συμβαίνει στο Xen.

## 3.2 Εικονικές Μηχανές (Virtual Machines)

Οι εικονικές μηχανές (virtual machines) είναι μια παραδοσιακή μέθοδος εικονοποίησης, όπου ένας hypervisor προσφέρει ένα ολοκληρωμένο εικονικό hardware περιβάλλον σε κάθε guest λειτουργικό σύστημα. Αυτό σημαίνει ότι το guest OS πιστεύει πως «τρέχει» σε έναν πραγματικό υπολογιστή, με τη δική του CPU, τη δική του μνήμη, τους δίσκους του και άλλα περιφερειακά.

### 3.2.1 Δομή και λειτουργία

Σε ένα τυπικό σενάριο, ο hypervisor εγκαθίσταται απευθείας πάνω στο φυσικό υλικό (μοντέλο bare-metal). Παραδείγματα τέτοιων hypervisors είναι το VMware ESXi, το Microsoft Hyper-V και το Xen. Ο hypervisor διαχειρίζεται τους φυσικούς πόρους και τους κατανέμει στα διάφορα guest OS, τα οποία δεν έχουν άμεση πρόσβαση στο υλικό.

Εναλλακτικά, μπορεί κανείς να χρησιμοποιήσει έναν hypervisor που εγκαθίσταται πάνω από ένα υπάρχον λειτουργικό σύστημα (μοντέλο hosted), όπως το VirtualBox ή το VMware Workstation. Σε αυτή την περίπτωση, το host λειτουργικό σύστημα διαμεσολαβεί για την πρόσβαση στο υλικό, ενώ ο hypervisor λειτουργεί ως μια εφαρμογή που προσφέρει τις υπηρεσίες εικονοποίησης.

### 3.2.2 Πλεονεκτήματα και μειονεκτήματα

#### Πλεονεκτήματα

- **Απομόνωση:** Κάθε virtual machine λειτουργεί εντελώς ανεξάρτητα από τις υπόλοιπες. Τυχόν σφάλματα ή επιθέσεις σε ένα guest δεν επηρεάζουν τους άλλους.
- **Ευελιξία:** Μπορούν να εγκατασταθούν διαφορετικά λειτουργικά συστήματα (Windows, Linux, BSD κλπ.) στον ίδιο φυσικό server.
- **Φορητότητα:** Οι εικονικές μηχανές μπορούν εύκολα να μεταφερθούν ή να αντιγραφούν από έναν host σε άλλο, επιτρέποντας απρόσκοπτη μετεγκατάσταση και ανάκτηση από βλάβες (disaster recovery).

#### Μειονεκτήματα

- **Υψηλότερη κατανάλωση πόρων:** Εφόσον κάθε guest OS «κουβαλά» το δικό του πυρήνα και ολοκληρωμένες βιβλιοθήκες, η κατανάλωση μνήμης και επεξεργαστικής ισχύος είναι σημαντική.
- **Χρόνοι εκκίνησης:** Η εκκίνηση μιας virtual machine απαιτεί τη φόρτωση πλήρους λειτουργικού συστήματος, γεγονός που αυξάνει τους χρόνους εκκίνησης σε σχέση με άλλα μοντέλα εικονοποίησης.



### 3.2.3 Χρήσεις και παραδείγματα

- **Παροχή υπηρεσιών cloud:** Πλατφόρμες όπως το Amazon EC2, το Microsoft Azure και το Google Cloud ξεκίνησαν προσφέροντας virtual machines, δίνοντας τη δυνατότητα στους πελάτες να «ενοικιάζουν» υπολογιστική ισχύ, εγκαθιστώντας ελεύθερα ό,τι λογισμικό επιθυμούν.
- **Απομόνωση εφαρμογών υψηλής ασφαλείας:** Συχνά, κρίσιμες εφαρμογές ή περιβάλλοντα δοκιμών (testing environments) φιλοξενούνται σε ξεχωριστές virtual machines για να εξασφαλίζεται η ασφάλεια των δεδομένων.
- **Περιβάλλοντα ανάπτυξης:** Οι προγραμματιστές αξιοποιούν εργαλεία όπως το VirtualBox ή το VMware Workstation για να δοκιμάζουν πολλαπλά λειτουργικά συστήματα στην ίδια φυσική μηχανή.

## 3.3 Containers

Τα containers (ή «κοντέινερς») αποτελούν μια πιο ελαφριά μορφή εικονοποίησης. Αντί να δημιουργείται ένα πλήρες εικονικό hardware περιβάλλον για κάθε guest, τα containers μοιράζονται τον ίδιο πυρήνα (kernel) του λειτουργικού συστήματος, ενώ απομονώνουν τις διεργασίες και τις βιβλιοθήκες που απαιτούνται για την εκτέλεση εφαρμογών.

### 3.3.1 Βασικές αρχές και τεχνολογίες

Το κλειδί για τη λειτουργία των containers είναι η αξιοποίηση των δυνατοτήτων απομόνωσης (namespaces) και ελέγχου πόρων (cgroups) που προσφέρουν οι μοντέρνοι πυρήνες Linux. Αυτά επιτρέπουν σε κάθε container να διατηρεί το δικό του χώρο διεργασιών, δικτύου και συστήματος αρχείων, χωρίς να γνωρίζει την ύπαρξη άλλων containers.

Μία από τις πιο διαδεδομένες πλατφόρμες για τη δημιουργία και τη διαχείριση containers είναι το Docker, το οποίο έχει γίνει συνώνυμο με την ορολογία των containers. Άλλες σχετικές τεχνολογίες περιλαμβάνουν το LXC (Linux Containers), το Podman και πλατφόρμες εντοπισμού όπως το Kubernetes.

### 3.3.2 Πλεονεκτήματα και μειονεκτήματα

#### Πλεονεκτήματα

- **Ελαφρότητα:** Επειδή οι containers δεν περιέχουν ολόκληρο λειτουργικό σύστημα, το αποτύπωμά τους σε CPU, μνήμη και αποθηκευτικό χώρο είναι σημαντικά μικρότερο.
- **Ταχύτητα:** Η εκκίνηση ενός container είναι ταχύτερη, καθώς δεν απαιτείται φόρτωση νέου πυρήνα. Επιπλέον, η ανάπτυξη νέων εκδόσεων εφαρμογών γίνεται με μεγαλύτερη ευκολία.
- **Ευκολία μεταφοράς (portability):** Τα containers προσφέρουν ένα συνεπές περιβάλλον εκτέλεσης, επιτρέποντας στις εφαρμογές να «τρέχουν» χωρίς αλλαγές σε διαφορετικά συστήματα Linux, Windows ή macOS (με τη βοήθεια ειδικών μηχανισμών).

#### Μειονεκτήματα

- **Κοινός πυρήνας (kernel):** Όλοι οι containers μοιράζονται τον ίδιο πυρήνα. Επομένως, αν εντοπιστεί ένα κενό ασφαλείας σε αυτόν, επηρεάζονται όλα τα containers.

- **Περιορισμοί σε OS-level διαφοροποιήσεις:** Δεδομένου ότι δεν υπάρχει ξεχωριστός πυρήνας, οι containers πρέπει να είναι συμβατοί με την έκδοση του πυρήνα του host. Αυτό περιορίζει τη δυνατότητα εκτέλεσης διαφορετικών λειτουργικών συστημάτων (π.χ. Windows σε Linux μηχανή).

### 3.3.3 Χρήσεις και παραδείγματα

- **Microservices:** Οι σύγχρονες αρχιτεκτονικές microservices βασίζονται συχνά σε containers, αφού κάθε υπηρεσία πακετάρεται με τις βιβλιοθήκες της και τρέχει ανεξάρτητα.
- **Continuous Integration/Continuous Deployment (CI/CD):** Εργαλεία όπως το Jenkins, το GitLab CI και το GitHub Actions υποστηρίζουν Docker containers για να εξασφαλίσουν επαναληψιμότητα και σταθερή συμπεριφορά κατά τη φάση δοκιμών και ανάπτυξης.
- **Serverless υποδομές:** Αν και τα serverless περιβάλλοντα κρύβουν την εσωτερική αρχιτεκτονική, πολλά από αυτά βασίζονται σε containers για την εκτέλεση των λειτουργιών (functions) των χρηστών.

## 3.4 Σύγκριση Virtual Machines και Containers

Παρά το γεγονός ότι αμφότερες οι τεχνολογίες εντάσσονται στο πλαίσιο της εικονοποίησης (virtualization), υπάρχουν ουσιώδεις διαφορές μεταξύ virtual machines και containers. Στον Πίνακα 3.1 παρουσιάζονται συνοπτικά ορισμένες βασικές διαφορές.

Χαρακτηριστικό	Virtual Machines	Containers
<b>Απομόνωση</b>	Πλήρης απομόνωση μέσω hypervisor	Απομόνωση σε επίπεδο OS (namespaces, cgroups)
<b>Μέγεθος</b>	Μεγαλύτερο, αφού περιέχει ολόκληρο OS	Μικρότερο, μόνο οι βιβλιοθήκες και οι εξαρτήσεις της εφαρμογής
<b>Κατανάλωση πόρων</b>	Υψηλότερη	Χαμηλότερη
<b>Εκκίνηση</b>	Αργή, απαιτεί φόρτωση λειτουργικού συστήματος	Γρήγορη, καθώς μοιράζεται τον πυρήνα του host
<b>Διαφορετικά λειτουργικά συστήματα</b>	Δυνατή εκτέλεση Windows, Linux, κλπ. στον ίδιο host	Όλοι οι containers πρέπει να είναι συμβατοί με τον πυρήνα του host

Πίνακας 3.1: Συνοπτική σύγκριση Virtual Machines και Containers

Από τα παραπάνω γίνεται σαφές ότι οι virtual machines είναι μια πιο «βαριά» αλλά και πιο γενική λύση, αφού υποστηρίζουν πλήρως διαφορετικά λειτουργικά συστήματα και αυξημένα επίπεδα απομόνωσης. Αντίθετα, τα containers προσφέρουν ταχύτητα, ευελιξία και ελαφρότητα, με βασική προϋπόθεση ότι μοιράζονται κοινό πυρήνα λειτουργικού συστήματος.

## 3.5 Προκλήσεις και τάσεις

### 3.5.1 Ασφάλεια (Security)

Οι ανησυχίες γύρω από την ασφάλεια αυξάνονται καθώς όλο και περισσότερες υποδομές βασίζονται σε εικονικά περιβάλλοντα. Στις virtual machines, ένας επιτιθέμενος που αποκτά πρόσβαση

σε μια guest μπορεί θεωρητικά να προσπαθήσει να «αποδράσει» στο επίπεδο του hypervisor, ενώ στους containers, η κοινή χρήση του πυρήνα καθιστά τον έλεγχο ασφάλειας του πυρήνα απολύτως κρίσιμο. Γι' αυτό, οι διαχειριστές φροντίζουν να παρακολουθούν εκδόσεις πυρήνων, ενημερώσεις λογισμικού και πολιτικές πρόσβασης (AppArmor, SELinux κ.λπ.) για να περιορίσουν τους κινδύνους.

### **3.5.2 Διαχείριση κλίμακας (Scalability)**

Η μαζική ανάπτυξη εκατοντάδων ή χιλιάδων containers ή virtual machines σε περιβάλλοντα cloud επιτάσσει εξελιγμένες τεχνικές αυτοματισμού και ενορχήστρωσης (orchestration). Τα εργαλεία Kubernetes, Docker Swarm και Mesos παρέχουν δυνατότητες αυτόματης διαχείρισης (autoscaling), παρακολούθησης (monitoring) και κατανομής φόρτου (load balancing), καθιστώντας τη λειτουργία σύνθετων συστημάτων βιώσιμη σε μεγάλη κλίμακα.



## 4. Kubernetes

Το Kubernetes αποτελεί μία από τις πιο δημοφιλείς και ολοκληρωμένες πλατφόρμες ενορχήστρωσης (orchestration) και διαχείρισης εφαρμογών που εκτελούνται σε περιβάλλοντα containers, αλλά και σε συνδυασμό με virtual machines. Αναπτύχθηκε αρχικά από την Google και προσφέρθηκε ως έργο ανοιχτού κώδικα (open-source), με ισχυρή υποστήριξη από την κοινότητα. Σήμερα, έχει εξελιχθεί σε έναν από τους βασικότερους πυλώνες για την ανάπτυξη καταναμημένων συστημάτων μεγάλης κλίμακας (distributed systems) και υποδομών cloud.

Στο πλαίσιο των προηγούμενων κεφαλαίων, όπου εξετάστηκαν το IoT (Internet of Things), τα CPS (Cyber-Physical Systems) και οι τεχνολογίες εικονοποίησης (virtual machines και containers), το παρόν κεφάλαιο αναδεικνύει τον ρόλο του Kubernetes ως ενοποιημένη πλατφόρμα υλοποίησης, ενορχήστρωσης και διαχείρισης αυτών των υποσυστημάτων. Αναλυτικότερα, θα παρουσιαστούν οι βασικές αρχές του Kubernetes, η αρχιτεκτονική του, οι δυνατότητες ασφαλο-ύς και κλιμακούμενης (scalable) ανάπτυξης εφαρμογών, καθώς και η σύνδεσή του με τις έννοιες του IoT, των CPS και των VMs.

### 4.1 Βασικές αρχές του Kubernetes

Το Kubernetes ακολουθεί μια αρχιτεκτονική client-server, όπου ένα κεντρικό control plane αναλαμβάνει τον έλεγχο πολλών worker nodes. Στο control plane περιλαμβάνονται διάφορες κρίσιμες υπηρεσίες, όπως ο API server, ο scheduler και οι controllers, ενώ στους worker nodes εκτελούνται τα πραγματικά φορτία εργασίας (workloads), δηλαδή οι containers ή οι virtual machines.

- **API server:** Παρέχει το κεντρικό σημείο επικοινωνίας μέσω REST API, μέσω του οποίου οι χρήστες, τα controllers και ο scheduler αλληλεπιδρούν με το Kubernetes cluster.
- **etcd:** Αποθηκεύει την κατάσταση (state) του cluster, π.χ. τις ρυθμίσεις (configurations) και τις προδιαγραφές των εφαρμογών.
- **scheduler:** Αποφασίζει σε ποιον worker node θα εκτελεστούν οι pods, βάσει διαθεσιμότητας πόρων (CPU, μνήμη) και διαφόρων πολιτικών.
- **kube-controller-manager:** Ελέγχουν και συντονίζουν τη «στοχευμένη» κατάσταση (desired state) με την «τρέχουσα» κατάσταση (current state). Για παράδειγμα, ένας controller εξασφαλίζει πως αν δηλωθούν 5 αντίγραφα (replicas) μιας εφαρμογής, θα εκτελούνται πάντοτε 5 pods.

Από την άλλη πλευρά, κάθε worker node φιλοξενεί:

- Τον **kubelet**, ο οποίος επικοινωνεί με τον API server και αναλαμβάνει την υλοποίηση των εντολών στο τοπικό σύστημα.
- Τον **kube-proxy**, υπεύθυνο για τους κανόνες δικτύωσης και την προώθηση (forwarding) της κίνησης στα σωστά pods.

- Τον **container runtime** (Docker, containerd ή άλλο) για τη δημιουργία και εκτέλεση containers.

## 4.2 Pods, Services και λοιποί πυρήνες μηχανισμοί

Στο Kubernetes, η μικρότερη μονάδα ανάπτυξης εφαρμογών είναι το pod. Ένα pod μπορεί να περιέχει έναν ή περισσότερους containers, οι οποίοι μοιράζονται το ίδιο namespace δικτύου και δίσκου (volumes). Πάνω από τα pods, ορίζονται έννοιες υψηλότερου επιπέδου, όπως το Deployment (εξασφαλίζει κλιμάκωση και ενημερώσεις χωρίς διακοπή) και το StatefulSet (για εφαρμογές που διατηρούν κατάσταση).

Για τη δικτύωση, το Kubernetes χρησιμοποιεί τους Services, οι οποίοι προσφέρουν ένα σταθερό σημείο πρόσβασης (cluster IP ή load balancer IP), ανεξάρτητα από το πού εκτελούνται τα pods φυσικά. Επίσης, υπάρχουν μηχανισμοί όπως τα Ingress controllers για την παροχή HTTPS τερματισμού (TLS termination) και δρομολόγησης (routing) σε υπηρεσίες εφαρμογών.

### 4.2.1 Αποθήκευση (Storage)

Για την επίμονη αποθήκευση δεδομένων (persistent data), το Kubernetes ορίζει τα Persistent Volumes (PV) και τα Persistent Volume Claims (PVC). Οι εφαρμογές (pods) ζητούν αποθηκευτικό χώρο δημιουργώντας ένα PVC, ενώ ο διαχειριστής (ή ένας μηχανισμός dynamic provisioning) αντιστοιχίζει ένα PV στο αίτημα αυτό. Έτσι, οι εφαρμογές μπορούν να κρατούν δεδομένα ανεξαρτήτως του node όπου εκτελούνται.

### 4.2.2 Ασφάλεια (Security)

Το Kubernetes διαθέτει πολλαπλά επίπεδα ασφάλειας:

- **Role-Based Access Control (RBAC):** Καθορίζει ποιοι χρήστες και ποιες υπηρεσίες έχουν δικαίωμα να εκτελούν συγκεκριμένες ενέργειες στο API.
- **Network Policies:** Ορίζουν κανόνες για τη δικτυακή ροή (ingress, egress) σε επίπεδο pod, εμποδίζοντας τη μη εξουσιοδοτημένη επικοινωνία.
- **Pod Security:** Επιτρέπει ή απαγορεύει τη χρήση ευαίσθητων προνομίων (privileged, root) ή την πρόσβαση σε κρίσιμα τμήματα του λειτουργικού συστήματος.

## 4.3 Kubernetes και ενοποίηση με containers, VMs, IoT και CPS

### 4.3.1 Containers και Kubernetes

Η πιο διαδεδομένη χρήση του Kubernetes αφορά την ενορχήστρωση εφαρμογών που εκτελούνται σε containers. Ανάμεσα στα δημοφιλέστερα container runtimes συναντάμε το Docker και το containerd. Το Kubernetes διευκολύνει:

- **Αυτόματη κλιμάκωση (autoscaling):** Αύξηση ή μείωση του αριθμού pods βάσει μετρικών (metrics), όπως η χρήση CPU ή οι αιτήσεις HTTP.
- **Αναβαθμίσεις χωρίς διακοπή (rolling updates):** Σταδιακή αντικατάσταση των pods με νέες εκδόσεις, διασφαλίζοντας τη συνεχή παροχή υπηρεσίας.

- **Ανθεκτικότητα** (resilience): Σε περίπτωση σφάλματος κάποιου pod ή ολόκληρου node, το Kubernetes επανεκκινεί αυτόματα τα pods σε υγιείς κόμβους.

### 4.3.2 Virtual Machines και KubeVirt

Παρά την έντονη έμφαση στους containers, κάποιες εφαρμογές ή περιπτώσεις χρήσης απαιτούν virtual machines. Το KubeVirt είναι ένα έργο ανοιχτού κώδικα που επιτρέπει την εκτέλεση VMs μέσα σε Kubernetes. Αυτό επιτυγχάνεται με την εισαγωγή CRDs (Custom Resource Definitions), μέσω των οποίων ορίζεται ένα VirtualMachine αντικείμενο στο Kubernetes API.

Με αυτή τη μέθοδο:

- **Υπάρχουσες VM-βασισμένες εφαρμογές** μπορούν να ενταχθούν στο Kubernetes οικοσύστημα, αξιοποιώντας τα οφέλη του (autoscaling, monitoring, logging).
- **Μεταβατικά σενάρια** μεταξύ VMs και containers γίνονται εφικτά, χωρίς να αλλάζει δραστικά η αρχιτεκτονική των συστημάτων.
- **Κοινή διαχείριση** πολιτικών, όπως η ασφάλεια και η δικτύωση, ενοποιείται για pods και VMs.

### 4.3.3 IoT και Edge Computing

Στα περιβάλλοντα IoT, η καταναεμημένη επεξεργασία δεδομένων (edge computing) αποκτά ιδιαίτερη σημασία, καθώς μειώνει τους χρόνους απόκρισης (latency) και το κόστος δικτυακής μετάδοσης. Εργαλεία όπως το k3s και το MicroK8s επιτρέπουν την εγκατάσταση ενός ελαφρού Kubernetes distribution σε edge devices με περιορισμένους πόρους. Κατ' αυτόν τον τρόπο, οι IoT συσκευές (sensors, actuators) μπορούν να επικοινωνούν με τοπικούς κόμβους, οι οποίοι τρέχουν συλλογισμούς (analytics, filtering) σε containers εντός ενός μικρού Kubernetes cluster.

Παράλληλα, τα δεδομένα που προκύπτουν από την τοπική επεξεργασία μπορούν να προωθηθούν στο κεντρικό cloud Kubernetes cluster, το οποίο αναλαμβάνει περαιτέρω ανάλυση (machine learning, big data analytics) και αποθήκευση (data warehousing). Αυτή η λογική edge-cloud επιτρέπει στα IoT συστήματα να επεκταθούν πιο εύκολα, να αυτοματοποιηθούν και να κλιμακώσουν τις υπηρεσίες τους χωρίς να επιβαρύνουν τους κεντρικούς πόρους με μη επεξεργασμένα δεδομένα.

### 4.3.4 Cyber-Physical Systems (CPS)

Τα CPS χαρακτηρίζονται από την αλληλεπίδραση μεταξύ φυσικών διεργασιών και ψηφιακών υποδομών. Το Kubernetes βοηθάει σημαντικά στην αρχιτεκτονική των CPS, ιδιαίτερα όταν χρησιμοποιούνται επιμέρους υπηρεσίες (microservices) για τη συλλογή και επεξεργασία σημάτων από αισθητήρες ή τον έλεγχο ενεργοποιητών (actuators).

Σε ορισμένες περιπτώσεις, τα CPS έχουν απαιτήσεις χαμηλής καθυστέρησης (low-latency) ή πραγματικού χρόνου (real-time). Παρότι το Kubernetes δεν σχεδιάστηκε για hard real-time εφαρμογές, υπάρχουν τροποποιημένοι πυρήνες Linux (RT kernels) και ειδικές ρυθμίσεις scheduler που μπορούν να βελτιώσουν την απόδοση. Έτσι, για συστήματα soft real-time ή ημι-αυτόνομες βιομηχανικές ροές, το Kubernetes μπορεί να προσφέρει μια ευέλικτη λύση ενορχήστρωσης και διαχείρισης.

## 4.4 Κλιμακούμενη ανάπτυξη και βέλτιστες πρακτικές

### 4.4.1 Autoscaling μηχανισμοί

Το Kubernetes υποστηρίζει διάφορες στρατηγικές αυτόματης κλιμάκωσης (autoscaling):

- **Horizontal Pod Autoscaler (HPA):** Αυξομειώνει τον αριθμό των pods ενός Deployment ή ReplicaSet βάσει μετρικών (CPU / memory χρήση ή custom metrics).
- **Cluster Autoscaler:** Αυτόματη προσθήκη ή αφαίρεση nodes σε ένα cloud περιβάλλον, αναλόγως με τις ανάγκες των pods.

Σε περιπτώσεις IoT ή CPS, όπου η ροή δεδομένων μπορεί να παρουσιάζει μεγάλες διακυμάνσεις, οι μηχανισμοί αυτοί εξασφαλίζουν την ομαλή ανταπόκριση του συστήματος.

### 4.4.2 Rolling Updates και Canary Deployments

Για τη συνεχή παράδοση (continuous delivery) αναβαθμίσεων χωρίς να διακόπτεται η λειτουργία, το Kubernetes χρησιμοποιεί rolling updates, αντικαθιστώντας σταδιακά τα παλιά pods με νέα. Επιπλέον, τεχνικές canary deployment επιτρέπουν τη δοκιμή νέων εκδόσεων σε μικρό ποσοστό χρηστών, προτού γενικευθούν σε όλο το σύστημα. Αυτό είναι ιδιαίτερα χρήσιμο όταν ενσωματώνονται αλγόριθμοι machine learning ή νέες λειτουργίες σε ευαίσθητα συστήματα CPS.

### 4.4.3 Παρακολούθηση (Monitoring) και καταγραφή (Logging)

Η διαχείριση σύνθετων Kubernetes clusters απαιτεί κατάλληλα εργαλεία παρακολούθησης (monitoring) και καταγραφής (logging). Συνήθως, χρησιμοποιούνται λύσεις όπως το Prometheus για τη συλλογή μετρικών, σε συνδυασμό με τον Grafana για την οπτικοποίηση. Για την ενιαία καταγραφή (logging), δημοφιλή είναι τα EFK (Elasticsearch, Fluentd, Kibana) ή το Loki της Grafana Labs. Με αυτόν τον τρόπο, οι διαχειριστές εντοπίζουν γρήγορα σφάλματα ή δυσλειτουργίες, τόσο σε εφαρμογές containers όσο και σε VMs (μέσω KubeVirt).



## 5. Αξιοποίηση Κυβερνοφυσικών Υπηρεσιών

Σε αυτό το κεφάλαιο παρουσιάζεται μια λεπτομερής ανάλυση του αυτοματοποιημένου συστήματος παραγωγής που χρησιμοποιείται για την κατασκευή των καρεκλών γραφείου τύπου Gregor. Επικεντρώνεται στην αρχιτεκτονική του συστήματος, τις βασικές λειτουργίες και τα ρομποτικά στοιχεία που επιτρέπουν την αυτόματη συναρμολόγηση με υψηλή ακρίβεια και επαναληψιμότητα. Θα εξετάσουμε τις διάφορες διαδικασίες που εμπλέκονται, καθώς και το πώς το σύστημα διαχειρίζεται την ενσωμάτωση των διαφορετικών συναρμολογήσεων για να παράγει ένα τελικό προϊόν που συμμορφώνεται με τα υψηλά πρότυπα ποιότητας.

### 5.1 Επισκόπηση Συστήματος

Το σύστημα αυτοματοποιημένης παραγωγής που αναπτύχθηκε για την κατασκευή των καρεκλών γραφείου τύπου Gregor χαρακτηρίζεται από μια σειρά καλά δομημένων σταδίων συναρμολόγησης, τα οποία διαχειρίζονται με ακρίβεια από εξειδικευμένα ρομπότ. Κάθε ρομπότ είναι ανατεθειμένο με συγκεκριμένες εργασίες, σχεδιασμένες για να εκτελούνται με βάση τη διαθεσιμότητα και την κατάσταση των εξαρτημάτων στις εργασιακές θέσεις.

Οι διαδικασίες παραγωγής ξεκινούν με τη συναρμολόγηση των βασικών δομικών στοιχείων, όπως είναι οι βάσεις και τα πόδια των καρεκλών. Το Ρομπότ 1 αναλαμβάνει την εκτέλεση του αρχικού σταδίου συναρμολόγησης, τοποθετώντας και ενώνοντας τα πόδια με τη βάση. Στη συνέχεια, το Ρομπότ 2 εισέρχεται για να προσθέσει επιπλέον δομικά στοιχεία όπως τα ροδάκια και τον μηχανισμό ανύψωσης, ολοκληρώνοντας το σκελετό της καρέκλας.

Η διαδικασία συνεχίζεται με τη συναρμολόγηση του καθίσματος και της πλάτης, που γίνεται σε χωριστές φάσεις και απαιτεί την εναλλαγή των ρομπότ ανάλογα με το στάδιο που βρίσκεται η καρέκλα στην παραγωγική γραμμή. Η προσθήκη των μπράτσων και της τελικής συναρμολόγησης είναι επιφορτισμένη στο Ρομπότ 3, το οποίο διαχειρίζεται τα τελευταία στάδια της διαδικασίας με μεγάλη ακρίβεια.

Κάθε ρομπότ υποβάλλει αιτήματα για να λάβει πρόσβαση στις εργασιακές θέσεις των πάγκων και διασφαλίζει ότι τα εξαρτήματα βρίσκονται στην κατάλληλη κατάσταση πριν ξεκινήσει η κάθε διαδικασία. Αυτό το σύστημα επιτρέπει την αποφυγή λαθών και συγκρούσεων, βελτιώνοντας τη συνολική αποδοτικότητα και ποιότητα της παραγωγής.

### 5.2 Στάδια Παραγωγής

Η διαδικασία παραγωγής των καρεκλών γραφείου τύπου Gregor διαχωρίζεται σε καθορισμένα στάδια που εξασφαλίζουν την ακριβή και αποδοτική συναρμολόγηση κάθε μέρους της καρέκλας. Αρχικά, το στάδιο S1 υποδέχεται την εναρκτήρια δράση από το Ρομπότ 1, το οποίο συναρμολογεί τα πόδια με την κεντρική βάση, δημιουργώντας τον πρωταρχικό σκελετό της καρέκλας. Συνεχίζοντας, στο στάδιο S2, το Ρομπότ 2 προσθέτει τα ροδάκια και τον μηχανισμό ανύψωσης, ενισχύοντας τη βάση για καλύτερη κινητικότητα και ρυθμιζόμενη στήριξη.

Η συνέχεια της κατασκευής μεταφέρεται ξανά στο Ρομπότ 1, το οποίο στο στάδιο S3 προετοιμάζει το κάθισμα ενσωματώνοντάς το με την πλάκα καθίσματος. Ακολούθως, στο στάδιο S4, το Ρομπότ 2 αναλαμβάνει να ενώσει το κάθισμα με την προετοιμασμένη βάση, ολοκληρώνοντας την ενσωμάτωση των κυριότερων μερών της καρέκλας.

Τα τελευταία στάδια εντάσσουν τη δράση του Ρομπότ 3, όπου το S5 αφορά την προσθήκη της πλάτης της καρέκλας. Στο S6 και S7, το ίδιο ρομπότ προσθέτει τα μπράτσα, το αριστερό και το δεξί αντίστοιχα, ενισχύοντας τη λειτουργικότητα και την άνεση του σχεδίου. Το τελικό στάδιο, S8, σηματοδοτεί την πλήρη ολοκλήρωση της καρέκλας, με την κατασκευή να έχει ολοκληρωθεί πλήρως και την καρέκλα να είναι έτοιμη για χρήση.

Η οργανωμένη διαδικασία και η σειριακή ολοκλήρωση των σταδίων διασφαλίζουν ότι κάθε μέρος της καρέκλας συναρμολογείται με ακρίβεια και αποδοτικότητα, αποφεύγοντας επαναλήψεις ή παραλείψεις στην παραγωγική διαδικασία.

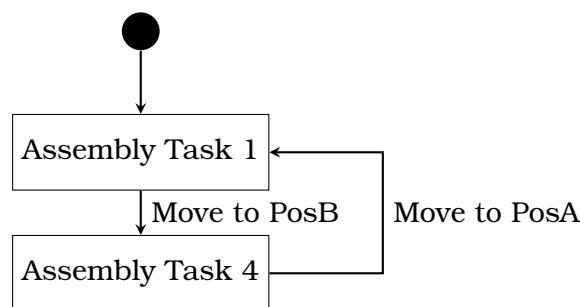
## 5.3 Ρόλοι και Ευθύνες

### 5.3.1 Ρομπότ 1

Το Ρομπότ 1 κατέχει κεντρικό ρόλο στην αυτοματοποιημένη γραμμή παραγωγής των καρεκλών γραφείου τύπου Gregor, με τις ευθύνες του να επικεντρώνονται σε κρίσιμες εργασίες συναρμολόγησης στην αρχική φάση της κατασκευής. Αρχικά, το Ρομπότ 1 εκτελεί την Εργασία Συναρμολόγησης 1 (AT1), όπου είναι υπεύθυνο για τη συναρμολόγηση της βάσης της καρέκλας, ενώνοντας τα πόδια με την κεντρική βάση στον πρώτο πάγκο εργασίας. Το αντίστοιχο διάγραμμα ροής του Ρομπότ 1 φαίνεται στο Σχήμα 5.1.

Κατόπιν, στην Εργασία Συναρμολόγησης 4 (AT4), το Ρομπότ 1 επιστρέφει για να ενσωματώσει το κάθισμα με την πλάκα καθίσματος, ρυθμίζοντας τα για την τελική τοποθέτηση. Η διαδικασία αυτή απαιτεί υψηλή ακρίβεια και είναι κρίσιμη για την άνεση και την ασφάλεια της τελικής κατασκευής.

Το Ρομπότ 1 έχει επίσης την ευθύνη να επαναλάβει τις εργασίες συναρμολόγησης 1 και 4 (AT1 και AT4) σε κυκλική διαδικασία, εξασφαλίζοντας τη συνεχή παραγωγή και αποδοτική λειτουργία της γραμμής παραγωγής. Αυτή η κυκλική και επαναληπτική δράση διασφαλίζει ότι η παραγωγή δεν διακόπτεται και ότι κάθε καρέκλα παράγεται με συνέπεια σύμφωνα με τα υψηλά πρότυπα ποιότητας του συστήματος.

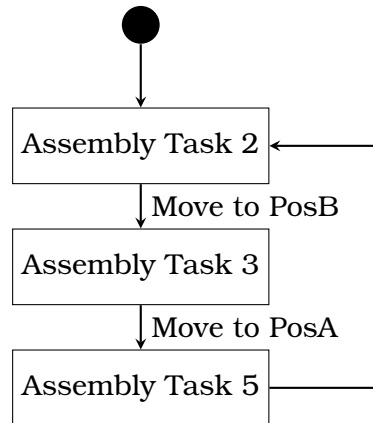


Σχήμα 5.1: Διάγραμμα Ροής Ρομπότ 1

### 5.3.2 Ρομπότ 2

Το Ρομπότ 2 εκτελεί ζωτικές λειτουργίες στην παραγωγή της καρέκλας Gregor, εστιάζοντας στη συναρμολόγηση βασικών εξαρτημάτων για τη λειτουργικότητα της καρέκλας. Το αντίστοιχο διάγραμμα ροής του Ρομπότ 2 φαίνεται στο Σχήμα 5.2.

Στην Εργασία Συναρμολόγησης 2 (AT2), το Ρομπότ 2 τοποθετεί τα ροδάκια και τον μηχανισμό ανύψωσης στην βάση της καρέκλας στον πρώτο πάγκο εργασίας. Στην Εργασία Συναρμολόγησης 3 (AT3), το Ρομπότ 2 αναλαμβάνει την ενσωμάτωση της πλάκας καθίσματος στην βάση, απαιτώντας ακρίβεια στην τοποθέτηση για να διασφαλιστεί η άνεση και σταθερότητα του καθίσματος. Τέλος, στην Εργασία Συναρμολόγησης 5 (AT5), το Ρομπότ 2 συναρμολογεί το τελικό κομμάτι της καρέκλας, ενώνοντας το κάθισμα με το υπόλοιπο σκελετό για να ολοκληρώσει τη δομή της καρέκλας.



Σχήμα 5.2: Διάγραμμα Ροής Ρομπότ 2

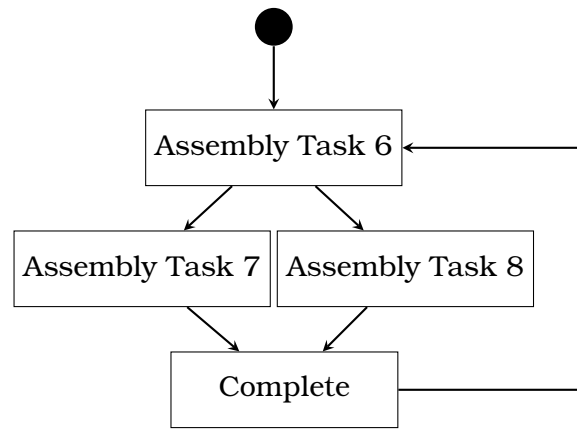
### 5.3.3 Ρομπότ 3

Το Ρομπότ 3 είναι αρμόδιο για την τελική φάση της συναρμολόγησης των καρεκλών γραφείου Gregor, ολοκληρώνοντας τη δομή με την προσθήκη κρίσιμων εξαρτημάτων για τη λειτουργικότητα και άνεση του τελικού προϊόντος. Το αντίστοιχο διάγραμμα ροής του Ρομπότ 3 φαίνεται στο Σχήμα 5.3.

Συγκεκριμένα, στην Εργασία Συναρμολόγησης 6 (AT6), το Ρομπότ 3 είναι υπεύθυνο για την τοποθέτηση της πλάτης της καρέκλας. Αυτή η εργασία απαιτεί ακρίβεια για να εξασφαλιστεί ότι η πλάτη στερεώνεται σωστά στον σκελετό, παρέχοντας την απαραίτητη στήριξη και άνεση.

Στα επόμενα στάδια, Εργασία Συναρμολόγησης 7 και Εργασία Συναρμολόγησης 8 (AT7 και AT8), το Ρομπότ 3 προχωρά στην τοποθέτηση των μπράτσων της καρέκλας, το αριστερό και το δεξί αντίστοιχα. Κάθε μπράτσο πρέπει να ενσωματωθεί με ακρίβεια για να διασφαλιστεί η άνεση και η σταθερότητα της καρέκλας, καθώς και η εργονομία για τον τελικό χρήστη.

Οι εργασίες που εκτελεί το Ρομπότ 3 είναι ουσιαστικές για την ολοκλήρωση της κατασκευής της καρέκλας, διασφαλίζοντας ότι το τελικό προϊόν είναι πλήρως λειτουργικό και άνετο. Η ακριβής εκτέλεση αυτών των τελευταίων βημάτων καθορίζει την ποιότητα και την εμφάνιση της καρέκλας, καταστρώνοντας τη βάση για μια επιτυχημένη παραγωγή.



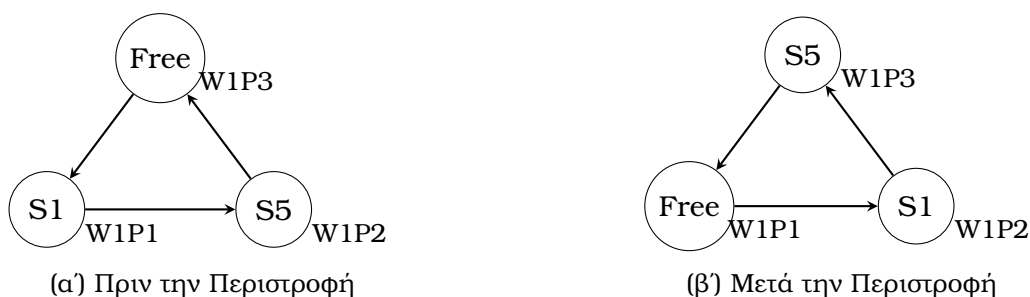
Σχήμα 5.3: Διάγραμμα Ροής Ρομπότ 3

### 5.3.4 Πάγκος Εργασίας 1

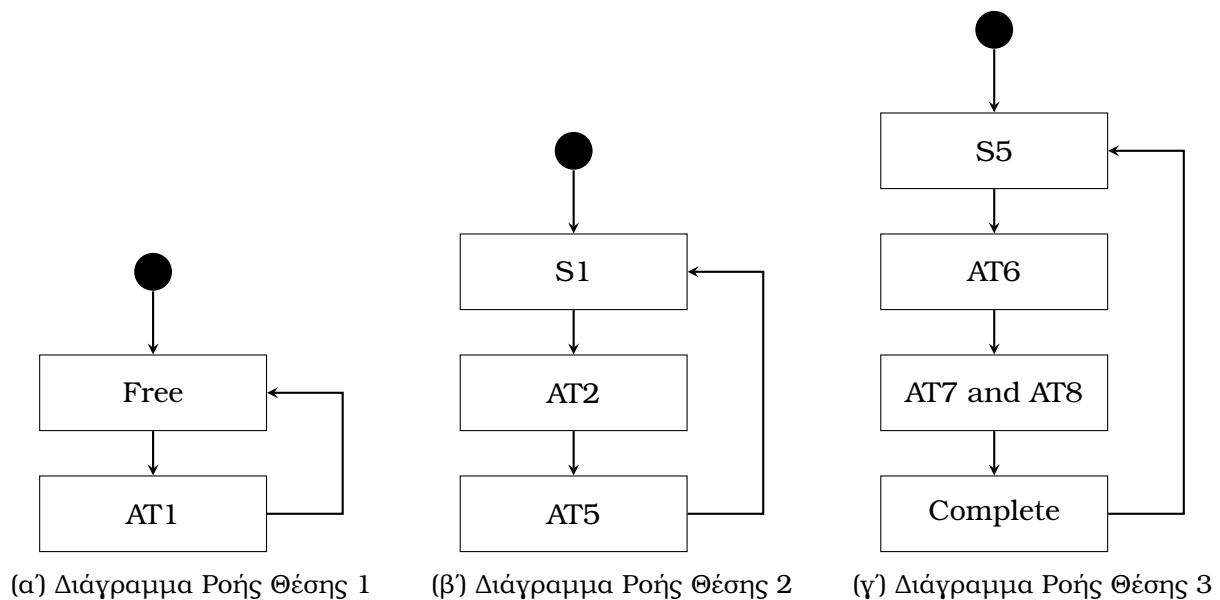
Ο Πάγκος Εργασίας 1 συνιστά ένα κρίσιμο στοιχείο στην αυτοματοποιημένη γραμμή παραγωγής των καρεκλών Gregor. Διαθέτει τρεις θέσεις εργασίας, οι οποίες ενεργοποιούνται συγκεκριμένα ανάλογα με τη φάση του προϊόντος που βρίσκεται σε καθεμιά. Το αντίστοιχα διαγράμματα ροής φαίνονται στα Σχήματα 5.4 και 5.5.

Η πρώτη θέση (W1P1) χρησιμοποιείται για την Εργασία Συναρμολόγησης 1 (AT1), όπου το Ρομπότ 1 συναρμολογεί τα πόδια με τη βάση της καρέκλας, δημιουργώντας το προϊόν στάδιο S1. Η δεύτερη θέση (W1P2) διαχειρίζεται την Εργασία Συναρμολόγησης 2 (AT2) και την Εργασία Συναρμολόγησης 5 (AT5), όπου το Ρομπότ 2 προσθέτει τα ροδάκια και τον μηχανισμό ανύψωσης (S2) και στη συνέχεια το κάθισμα στον σκελετό της καρέκλας, ολοκληρώνοντας το στάδιο S5. Η τρίτη θέση (W1P3) είναι προορισμένη για την Εργασία Συναρμολόγησης 6 (AT6), όπου το Ρομπότ 3 τοποθετεί την πλάτη της καρέκλας, φτάνοντας στο στάδιο S6.

Οι συνθήκες περιστροφής του Πάγκου Εργασίας 1 είναι καθοριστικές για την εκκίνηση των εργασιών. Μετά την πρώτη περιστροφή, η θέση W1P1 καθίσταται ελεύθερη, η W1P2 φιλοξενεί το προϊόν μετά το AT1 (S1) και η W1P3 είναι ελεύθερη, εκκινώντας την εργασία AT2. Για τις επόμενες περιστροφές, απαιτείται η W1P1 να έχει προϊόν μετά το AT1 (S1), η W1P2 να φιλοξενεί προϊόν μετά το AT5 (S5) και η W1P3 να είναι ελεύθερη, εκκινώντας τις εργασίες AT2 και AT6.



Σχήμα 5.4: Διάγραμμα Ροής Πάγκου Εργασίας 1

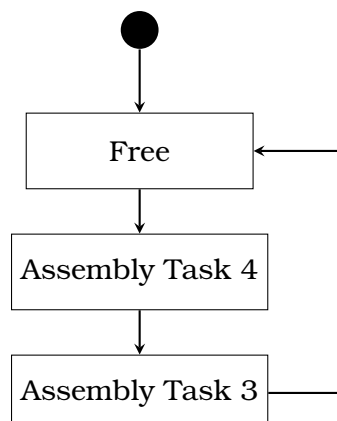


Σχήμα 5.5: Διάγραμμα Ροής Θέσεων Πάγκου Εργασίας 1

### 5.3.5 Πάγκος Εργασίας 2

Ο Πάγκος Εργασίας 2 στην παραγωγική γραμμή των καρεκλών Gregor είναι ένας εξειδικευμένος πάγκος με μοναδική θέση εργασίας, W2P1, η οποία διαχειρίζεται δύο κρίσιμες εργασίες συναρμολόγησης. Το Ρομπότ 1 χρησιμοποιεί αυτή τη θέση για την Εργασία Συναρμολόγησης 4 (AT4), όπου τοποθετείται το κάθισμα στη βάση του καθίσματος. Ακολούθως, το Ρομπότ 2 εκτελεί την Εργασία Συναρμολόγησης 3 (AT3) στην ίδια θέση, όπου συνδέει την πλάκα καθίσματος με τη βάση. Το αντίστοιχο διάγραμμα ροής του Πάγκου Εργασίας 2 φαίνεται στο Σχήμα 5.6.

Η ευθύνη του πάγκου εργασίας για τη διαχείριση του αιτήματος πρόσβασης από τα ρομπότ είναι καθοριστική, καθώς απαιτείται ο συντονισμός για την ορθή και έγκαιρη διαθεσιμότητα της θέσης για κάθε εργασία. Αυτός ο πάγκος εξασφαλίζει ότι οι συγκεκριμένες εργασίες εκτελούνται με ακρίβεια και στη σωστή σειρά, παρέχοντας μια αποτελεσματική βάση για την εκτέλεση εξειδικευμένων εργασιών που κρίνονται απαραίτητες για τη λειτουργικότητα και την ασφάλεια της τελικής καρέκλας.



Σχήμα 5.6: Διάγραμμα Ροής Πάγκου Εργασίας 2

## 5.4 Εργασίες Συναρμολόγησης

Στην ενότητα αυτή, θα εξετάσουμε λεπτομερώς τις διαδικασίες συναρμολόγησης των καρεκλών γραφείου τύπου Gregor, που αποτελούν κρίσιμο στοιχείο της αυτοματοποιημένης παραγωγικής γραμμής. Κάθε διαδικασία συναρμολόγησης έχει σχεδιαστεί για να εξασφαλίζει ακρίβεια και αποδοτικότητα, διασφαλίζοντας ότι το τελικό προϊόν πληροί τις υψηλές προδιαγραφές λειτουργικότητας που απαιτούνται.

Η διαδικασία συναρμολόγησης διαχωρίζεται σε διάφορα στάδια, με κάθε στάδιο να εμπλέκει την εκτέλεση συγκεκριμένων εργασιών από διαφορετικά ρομπότ. Αυτές οι εργασίες στοχεύουν στη σταδιακή κατασκευή της καρέκλας, από τη βάση μέχρι το τελικό κάθισμα με την πλάτη και τα μπράτσα. Η σωστή οργάνωση και η ακριβής ακολουθία των εργασιών είναι κρίσιμες για την ομαλή λειτουργία της γραμμής παραγωγής.

### 5.4.1 Εργασία Συναρμολόγησης 1

Η Εργασία Συναρμολόγησης 1 (AT1) αποτελεί το θεμελιώδες πρώτο στάδιο στη συναρμολόγηση των καρεκλών Gregor και διεξάγεται από το Ρομπότ 1 στην πρώτη θέση του Πάγκου Εργασίας 1 (W1P1). Αυτή η διαδικασία αποτελείται από τέσσερα βήματα και έχει ως στόχο την αρχική συναρμολόγηση της βάσης της καρέκλας.

1. **Βήμα 1:** Το Ρομπότ 1 αναλαμβάνει τα πόδια της καρέκλας από τον Χώρο Αποθήκευσης 1 και τα μεταφέρει προς τον πάγκο εργασίας.
2. **Βήμα 2:** Εφόσον λάβει άδεια πρόσβασης στη θέση W1P1, το ρομπότ τοποθετεί τα πόδια στην καθορισμένη θέση.
3. **Βήμα 3:** Στη συνέχεια, το Ρομπότ 1 επιστρέφει στον Χώρο Αποθήκευσης 2 για να προμηθευτεί την κεντρική βάση της καρέκλας και την τοποθετεί πάνω στα πόδια.
4. **Βήμα 4:** Κατά την τελική τοποθέτηση της βάσης, το Ρομπότ 1 εξασφαλίζει ότι όλες οι συνδέσεις είναι σταθερές και ακριβείς. Με την ολοκλήρωση αυτού του βήματος, ενεργοποιείται αυτόματα η Εργασία Συναρμολόγησης 4 (AT4).

Η AT1 εκκινείται αρχικά από το σύστημα κατά την έναρξη της παραγωγής και στη συνέχεια από την Εργασία Συναρμολόγησης 4 (AT4) κάθε φορά που ολοκληρώνεται. Αυτό διασφαλίζει ότι η αρχική βάση της καρέκλας συναρμολογείται σωστά και με την κατάλληλη σειρά, καθιστώντας το πρώτο στάδιο ζωτικής σημασίας για την ορθή εκκίνηση της διαδικασίας παραγωγής.

### 5.4.2 Εργασία Συναρμολόγησης 2

Η Εργασία Συναρμολόγησης 2 (AT2) πραγματοποιείται από το Ρομπότ 2 και εκτελείται στη δεύτερη θέση του Πάγκου Εργασίας 1 (W1P2). Αυτή η διαδικασία είναι απαραίτητη για την προσθήκη των ροδάκιων και του μηχανισμού ανύψωσης στην ήδη συναρμολογημένη βάση της καρέκλας. Η AT2 αποτελείται από τέσσερα βήματα:

1. **Βήμα 1:** Ενεργοποιείται όταν ο Πάγκος Εργασίας 1 περιστραφεί, φέρνοντας στη θέση W1P2 το αντικείμενο από την AT1. Το Ρομπότ 2 παίρνει τα ροδάκια από τον Χώρο Αποθήκευσης 4 και ζητά άδεια να εργαστεί στη θέση W1P2.
2. **Βήμα 2:** Ενεργοποιείται μόλις το Ρομπότ 2 λάβει άδεια για τη θέση W1P2. Συνδυάζει τα ροδάκια με το αντικείμενο σε αυτή τη θέση και ενεργοποιεί το επόμενο βήμα.

3. **Βήμα 3:** Συνεχίζεται με το Ρομπότ 2 να παίρνει τον μηχανισμό ανύψωσης από τον Χώρο Αποθήκευσης 5 και να ζητά ξανά άδεια για εργασία στη θέση W1P2.
4. **Βήμα 4:** Αφού λάβει άδεια, το Ρομπότ 2 συνδυάζει τον μηχανισμό ανύψωσης με το αντικείμενο στη θέση W1P2 και με την ολοκλήρωση αυτού του βήματος ενεργοποιεί την Εργασία Συναρμολόγησης 3 (AT3).

Η AT2 εκκινείται από την περιστροφή του Πάγκου Εργασίας 1 που φέρνει στη θέση W1P2 το αντικείμενο μετά την AT1. Το Ρομπότ 2 διαδραματίζει κεντρικό ρόλο στην επικύρωση και ενίσχυση της βάσης της καρέκλας, προετοιμάζοντας τη για τις επόμενες φάσεις συναρμολόγησης.

### 5.4.3 Εργασία Συναρμολόγησης 3

Η Εργασία Συναρμολόγησης 3 (AT3) εκτελείται από το Ρομπότ 2 και πραγματοποιείται κυρίως στον Πάγκο Εργασίας 2 (W2P1), ακολουθώντας την ολοκλήρωση της Εργασίας Συναρμολόγησης 4 (AT4). Αυτή η διαδικασία είναι κρίσιμη για την τελική σύνδεση και στερέωση της πλάκας καθίσματος με την κύρια συναρμολόγηση της καρέκλας. Η AT3 αποτελείται από τέσσερα βήματα:

1. **Βήμα 1:** Ξεκινά μετά την ολοκλήρωση της AT2, όπου το Ρομπότ 2 κινείται προς τη θέση B και ζητά άδεια για εργασία στον Πάγκο Εργασίας 2 (W2P1), όπου αναμένει το αντικείμενο μετά την AT4.
2. **Βήμα 2:** Ακολουθεί η ενεργοποίηση μετά την ολοκλήρωση της AT4 και τη χορήγηση άδειας από τον W2P1. Σε αυτό το σημείο, το Ρομπότ 2 συναρμολογεί την πλάκα καθίσματος με τη βάση, εξασφαλίζοντας τη στερέωση των δύο μερών.
3. **Βήμα 3:** Μετά τη συναρμολόγηση, το Ρομπότ 2 ζητά άδεια για να αναλάβει το αντικείμενο από τον W2P1 και κατευθύνεται προς τον Πάγκο Εργασίας 1 (W1P2).
4. **Βήμα 4:** Εφόσον λάβει άδεια για τη θέση W1P2, το Ρομπότ 2 τοποθετεί το αντικείμενο σε αυτή τη θέση και με την ολοκλήρωση αυτού του βήματος ενεργοποιεί την Εργασία Συναρμολόγησης 5 (AT5).

Η AT3 είναι ουσιαστική για την τελική φάση της κατασκευής του καθίσματος και την ένωση των κύριων συστατικών της καρέκλας. Αυτή η εργασία διασφαλίζει ότι το κάθισμα είναι σταθερά συνδεδεμένο με τη βάση, ετοιμάζοντας το προϊόν για τις τελευταίες ρυθμίσεις και προσθήκες πριν την τελική επιθεώρηση.

### 5.4.4 Εργασία Συναρμολόγησης 4

Η Εργασία Συναρμολόγησης 4 (AT4) είναι μια κρίσιμη διαδικασία στην παραγωγή των καρεκλών Gregor, εκτελείται από το Ρομπότ 1 και διαδραματίζει ρόλο στην τοποθέτηση και στερέωση της πλάκας του καθίσματος στην καρέκλα. Η διαδικασία πραγματοποιείται στην θέση του Πάγκου Εργασίας 2 (W2P1) και περιλαμβάνει τέσσερα βήματα:

1. **Βήμα 1:** Αρχίζει με την ενεργοποίηση του Ρομπότ 1 μετά την ολοκλήρωση της Εργασίας Συναρμολόγησης 1 (AT1). Το ρομπότ μετακινείται προς τη θέση B, αντιμετωπίζει τη μεταφορική ταινία 1 και παίρνει το αντικείμενο (κάθισμα). Στη συνέχεια, ζητά άδεια για να εργαστεί στη θέση W2P1.

2. **Βήμα 2:** Ενεργοποιείται μόλις το Ρομπότ 1 λάβει άδεια για τη θέση W2P1. Το ρομπότ τοποθετεί το κάθισμα στη θέση, ετοιμάζοντάς το για την προσθήκη της πλάκας του καθίσματος και ενεργοποιεί το επόμενο βήμα.
3. **Βήμα 3:** Συνεχίζεται με το Ρομπότ 1 να αντιμετωπίζει τον Χώρο Αποθήκευσης 3, απ' όπου παίρνει την πλάκα του καθίσματος και ζητά άδεια για να επισυνάψει το εξάρτημα στην ίδια θέση (W2P1).
4. **Βήμα 4:** Με τη λήψη άδειας, το Ρομπότ 1 συνδυάζει την πλάκα του καθίσματος με το κάθισμα στη θέση W2P1, συνδέοντας σταθερά τα δύο μέρη. Με την ολοκλήρωση της συναρμολόγησης, το ρομπότ μετακινείται πίσω στη θέση Α και ενεργοποιεί εκ νέου την Εργασία Συναρμολόγησης 1 (AT1), συνεχίζοντας την παραγωγική διαδικασία.

Η AT4 είναι ζωτικής σημασίας για τη στερέωση των κρίσιμων συστατικών του καθίσματος και διασφαλίζει τη συνέχιση της ομαλής και αποδοτικής παραγωγής της καρέκλας Gregor.

#### 5.4.5 Εργασία Συναρμολόγησης 5

Η Εργασία Συναρμολόγησης 5 (AT5) αποτελεί ένα κρίσιμο στάδιο στην τελική συναρμολόγηση της καρέκλας Gregor, όπου το κύριο σώμα της καρέκλας συνδέεται με το κάθισμα. Τη διαδικασία αυτή αναλαμβάνει το Ρομπότ 2 και εκτελείται στη δεύτερη θέση του Πάγκου Εργασίας 1 (W1P2). Η διαδικασία AT5 αποτελείται από δύο βήματα:

1. **Βήμα 1:** Ενεργοποιείται αμέσως μετά την ολοκλήρωση της Εργασίας Συναρμολόγησης 3 (AT3). Το Ρομπότ 2 ζητά άδεια για να εργαστεί στη θέση W1P2, όπου αναμένει το κύριο σώμα της καρέκλας μαζί με το προσαρμοσμένο κάθισμα.
2. **Βήμα 2:** Μόλις λάβει άδεια για τη θέση W1P2, το Ρομπότ 2 συνδυάζει τα δύο μέρη, χρησιμοποιώντας εξειδικευμένα εργαλεία για να σφίξει και να εξασφαλίσει τη στερέωσή τους. Αυτό το βήμα είναι κρίσιμο για τη σωστή λειτουργία της καρέκλας, καθώς η σταθερότητα της συνδέσεως επηρεάζει άμεσα την άνεση και την ασφάλεια του χρήστη.

Η εκκίνηση της AT5 γίνεται αυτόματα μετά την ολοκλήρωση της AT3, όπου το κάθισμα έχει ήδη προσαρμοστεί στην βάση του καθίσματος και τοποθετηθεί ενώπιον του Ρομπότ 2. Το Ρομπότ 2, με αυτόν τον τρόπο, ολοκληρώνει τη σύνδεση των κύριων μερών της καρέκλας, επιτυγχάνοντας μια σταθερή και λειτουργική δομή που είναι έτοιμη για τις τελικές προσθήκες, όπως η πλάτη και τα μπράτσα.

#### 5.4.6 Εργασία Συναρμολόγησης 6

Η Εργασία Συναρμολόγησης 6 (AT6) πραγματοποιείται από το Ρομπότ 3 και είναι κρίσιμη για την τοποθέτηση της πλάτης στη συναρμολογημένη καρέκλα. Αυτή η διαδικασία λαμβάνει χώρα στην τρίτη θέση του Πάγκου Εργασίας 1 (W1P3) και αποτελείται από δύο βήματα:

1. **Βήμα 1:** Ενεργοποιείται μετά από τη δεύτερη περιστροφή του Πάγκου Εργασίας 1, οπότε και η W1P3 πρέπει να περιέχει το αντικείμενο μετά την Εργασία Συναρμολόγησης 5 (AT5). Το Ρομπότ 3 αντιμετωπίζει τη μεταφορική ταινία 2 και παίρνει την πλάτη της καρέκλας. Στη συνέχεια, ζητά άδεια για να εργαστεί στη θέση W1P3.
2. **Βήμα 2:** Με τη λήψη άδειας, το Ρομπότ 3 τοποθετεί την πλάτη στο κύριο σώμα της καρέκλας και εξασφαλίζει τη στερέωσή της. Αυτό το βήμα απαιτεί ακριβή τοποθέτηση και στερέωση για να διασφαλιστεί η άνεση και η ασφάλεια του τελικού προϊόντος. Με



την ολοκλήρωση αυτού του βήματος, το Ρομπότ 3 ενεργοποιεί ταυτόχρονα τις Εργασίες Συναρμολόγησης 7 και 8 (AT7 και AT8), που αφορούν την τοποθέτηση των μπράτσων της καρέκλας.

Η AT6 είναι ζωτικής σημασίας για την τελική εμφάνιση και λειτουργικότητα της καρέκλας, καθώς η πλάτη παρέχει στήριξη και άνεση για τον χρήστη. Η εκκίνηση αυτής της διαδικασίας από την ενεργοποίηση του Πάγκου Εργασίας 1 διασφαλίζει την ορθή σειρά συναρμολόγησης και την ομαλή ενσωμάτωση της πλάτης στο συνολικό σχεδιασμό της καρέκλας.

#### 5.4.7 Εργασία Συναρμολόγησης 7

Η Εργασία Συναρμολόγησης 7 (AT7) είναι μία από τις τελικές φάσεις στην παραγωγή της καρέκλας Gregor, όπου το Ρομπότ 3 είναι υπεύθυνο για την τοποθέτηση του αριστερού μπράτσου στην καρέκλα. Αυτή η διαδικασία εκτελείται στην τρίτη θέση του Πάγκου Εργασίας 1 (W1P3) και αποτελείται από δύο βήματα:

1. **Βήμα 1:** Ενεργοποιείται ταυτόχρονα με την Εργασία Συναρμολόγησης 8 (AT8) αφού ολοκληρωθεί η Εργασία Συναρμολόγησης 6 (AT6). Το Ρομπότ 3 κινείται προς τον Χώρο Αποθήκευσης 6 και παίρνει το αριστερό μπράτσο της καρέκλας. Στη συνέχεια, ζητά άδεια για να εργαστεί στη θέση W1P3.
2. **Βήμα 2:** Με τη λήψη άδειας, το Ρομπότ 3 τοποθετεί το αριστερό μπράτσο στην καρέκλα, ενώνοντάς το με την υπόλοιπη κατασκευή. Η στερέωση πρέπει να είναι ακριβής και σταθερή, για να εγγυηθεί την άνεση και τη λειτουργικότητα του τελικού προϊόντος.

Η AT7 είναι ζωτική για την προσθήκη των τελευταίων λειτουργικών στοιχείων που καθορίζουν την άνεση και την εργονομική αξία της καρέκλας. Η ενσωμάτωση του μπράτσου γίνεται με προσοχή για να διασφαλιστεί η διαρκής αντοχή και η ασφάλεια κατά τη χρήση.

#### 5.4.8 Εργασία Συναρμολόγησης 8

Η Εργασία Συναρμολόγησης 8 (AT8) είναι ένα από τα τελικά στάδια στη διαδικασία συναρμολόγησης της καρέκλας Gregor, όπου το Ρομπότ 3 αναλαμβάνει την τοποθέτηση του δεξιού μπράτσου. Αυτή η διαδικασία πραγματοποιείται στην τρίτη θέση του Πάγκου Εργασίας 1 (W1P3) και ενεργοποιείται ταυτόχρονα με την Εργασία Συναρμολόγησης 7 (AT7) ακριβώς μετά την ολοκλήρωση της Εργασίας Συναρμολόγησης 6 (AT6). Η διαδικασία AT8 αποτελείται από τα εξής βήματα:

1. **Βήμα 1:** Το Ρομπότ 3 κινείται προς τον Χώρο Αποθήκευσης 6 και αναλαμβάνει το δεξί μπράτσο της καρέκλας. Εν συνεχεία, ζητά άδεια για εργασία στη θέση W1P3, όπου πρόκειται να προσαρμόσει το μπράτσο στο κύριο σώμα της καρέκλας.
2. **Βήμα 2:** Αφού λάβει την απαραίτητη άδεια, το Ρομπότ 3 τοποθετεί το δεξί μπράτσο στην καρέκλα, ενώνοντάς το με το σκελετό. Η σύνδεση πρέπει να γίνει με ακρίβεια για να διασφαλιστεί η στερέωση και η σωστή λειτουργία του τελικού προϊόντος.

Η ολοκλήρωση της AT8 εγγυάται ότι η καρέκλα είναι πλήρως συναρμολογημένη με τα μπράτσα, παρέχοντας την απαραίτητη στήριξη και ολοκληρώνοντας την κατασκευή της. Με την τοποθέτηση του τελευταίου μπράτσου, η καρέκλα είναι έτοιμη για τελική χρήση, προσφέροντας την απαιτούμενη στήριξη και σταθερότητα.



## **6. Υλοποίηση των υποσυστημάτων με χρήση Kubernetes**



## **7. Συμπεράσματα – Μελλοντική εργασία**



# Βιβλιογραφία

- [1] K. Thramboulidis, D. C. Vachtsevanou, and A. Solanos, «Cyber-physical microservices: An iot-based framework for manufacturing systems», in 2018 IEEE Industrial Cyber-Physical Systems (ICPS), IEEE, 2018, pp. 232-239.
- [2] J. Gubbi, R. Buyya, S. Marusic, and M. Palaniswami, «Internet of things (IoT): A vision, architectural elements, and future directions», Future Generation Computer Systems, vol. 29, no. 7, pp. 1645-1660, Sep. 1, 2013, issn: 0167-739X. doi: [10.1016/j.future.2013.01.010](https://doi.org/10.1016/j.future.2013.01.010). [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0167739X13000241>.
- [3] R. H. Weber, «Internet of things - new security and privacy challenges», Computer Law & Security Review, vol. 26, no. 1, pp. 23-30, Jan. 1, 2010, issn: 0267-3649. doi: [10.1016/j.clsr.2009.11.008](https://doi.org/10.1016/j.clsr.2009.11.008). [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0267364909001939>.
- [4] D. Evans, The internet of things: How the next evolution of the internet is changing everything, Cisco White Paper, 2011.
- [5] L. Atzori, A. Iera, and G. Morabito, «The internet of things: A survey», Computer Networks, vol. 54, no. 15, pp. 2787-2805, Oct. 28, 2010, issn: 1389-1286. doi: [10.1016/j.comnet.2010.05.010](https://doi.org/10.1016/j.comnet.2010.05.010). [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S1389128610001568>.
- [6] I. Cisco Systems, The internet of things reference model, Cisco White Paper, 2014.
- [7] E. A. Lee, «Cyber physical systems: Design challenges», in 11th IEEE International Symposium on Object Oriented Real-Time Distributed Computing, IEEE, 2008, pp. 363-369.
- [8] R. Rajkumar, I. Lee, L. Sha, and J. Stankovic, «Cyber-physical systems: The next computing revolution», in Design Automation Conference (DAC), ACM/IEEE, 2010, pp. 731-736.
- [9] R. Baheti and H. Gill, Cyber-physical systems, Online: <https://www.nsf.gov/pubs/2011/nsf11033/nsf11033.jsp>, 2011.