



**ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΙΡΑΙΩΣ**

**ΣΧΟΛΗ ΤΕΧΝΟΛΟΓΙΩΝ ΠΛΗΡΟΦΟΡΙΚΗΣ & ΕΠΙΚΟΙΝΩΝΙΩΝ**

**ΤΜΗΜΑ ΨΗΦΙΑΚΩΝ ΣΥΣΤΗΜΑΤΩΝ**

**Π.Μ.Σ. “ΠΛΗΡΟΦΟΡΙΑΚΑ ΣΥΣΤΗΜΑΤΑ & ΥΠΗΡΕΣΙΕΣ”**

**ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ**

Αυτοματοποίηση και συντονισμός διαδικασιών λογισμικού ως υπηρεσία (SaaS) από άκρο σε άκρο με έμφαση στην υλοποίηση υπηρεσιών.

(End-to-end SaaS process automation & orchestration with emphasis on services development)

Ον/μο Φοιτητή: Λαλιώτης Νικόλαος ( ΜΕ2141 )

Επιβλέπων Καθηγητής: Κυριαζής Δημοσθένης

Πειραιάς

Σεπτέμβριος 2023

## **ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ**

## **ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ**

Αυτοματοποίηση και συντονισμός διαδικασιών λογισμικού ως υπηρεσία (SaaS) από άκρο σε άκρο.

(End-to-end SaaS process automation & orchestration)

Λαλιώτης Νικόλαος (ME2141)

## Περίληψη

Η παρούσα εργασία επικεντρώνεται στην αυτοματοποίηση των διαδικασιών σε δημόσιους φορείς με την χρήση του Camunda. Αναλύονται τα βασικά στοιχεία του προτύπου BPMN (Business Process Model and Notation) και στην συνέχεια, παρουσιάζονται τα πλεονεκτήματα του Camunda ως ισχυρό εργαλείο για την αυτοματοποίηση επιχειρησιακών διαδικασιών. Παρέχονται λεπτομερείς οδηγίες για την εγκατάσταση του Camunda σε ένα περιβάλλον on-premises. Ακόμη, περιγράφεται η τεχνική υλοποίηση μιας εφαρμογής, συμπεριλαμβανομένων των διαδικασιών στο Camunda, του frontend και του Web API.

Τα αποτελέσματα της μελέτης εξετάζονται και αξιολογούνται, με έμφαση στην απόδοση και την αποτελεσματικότητα του συστήματος αυτοματισμού διεργασιών. Επιπλέον, παρουσιάζονται μελλοντικά βήματα για την ενδυνάμωση του συστήματος, με στόχο την ενίσχυση της αυτοματοποίησης διαδικασιών στο δημόσιο τομέα.

Συνοψίζοντας, η εργασία αποσκοπεί στην προώθηση της αυτοματοποίησης διαδικασιών στον δημόσιο τομέα, καθώς και την ενίσχυση της απόδοσης και της αποτελεσματικότητας των οργανισμών. Επιπλέον, το Camunda συνιστάται ως ένα πολύτιμο εργαλείο για την καινοτομία και την πρόοδο στον δημόσιο τομέα. Με την εφαρμογή των αυτοματοποιημένων διαδικασιών, οι φορείς μπορούν να επιτύχουν αυξημένη αποτελεσματικότητα, μείωση σφαλμάτων και βελτιωμένη ροή εργασιών, προάγοντας έτσι την καλύτερη παροχή υπηρεσιών προς τους πολίτες.

Βασικές Έννοιες: αυτοματοποίηση επιχειρησιακών διαδικασιών, μοντελοποίηση, βελτίωση συστήματος, πρότυπο BPMN, πλατφόρμα Camunda

## Abstract

This thesis focuses on process automation in public organizations using Camunda. It analyzes the key elements of the BPMN (Business Process Model and Notation) standard and presents the advantages of Camunda as a powerful tool for process automation. The thesis provides detailed instructions for installing Camunda in an on-premises environment and describes the technical implementation of an application, including the processes in Camunda, the frontend, and the web API.

The results of the study are examined and evaluated, with a focus on the performance and effectiveness of the process automation system. Furthermore, future plans for further enhancing the system are presented, aiming to strengthen process automation in the public sector.

In summary, the thesis aims to promote process automation in the public sector and enhance the performance and effectiveness of organizations. Additionally, Camunda is recommended as a valuable tool for innovation and progress in the public sector. By implementing automated processes, organizations can achieve increased efficiency, reduced errors, and improved workflow, thereby facilitating better service delivery to citizens.

Keywords: process automation, modeling, improvement of system, BPMN, Camunda platform

*"Η μόρφωση είναι ο πυλώνας της ελευθερίας. Σε οποιονδήποτε τομέα και αν κινηθείς, επένδυσε στη γνώση και η πρόοδος θα ακολουθήσει." - John F. Kennedy*

## Ευχαριστίες

Θα ήθελα να εκφράσω τις βαθιές μου ευχαριστίες για την υποστήριξη και την εμπιστοσύνη που μου παραχωρήθηκαν κατά τη διάρκεια της συγγραφής της διπλωματικής μου εργασίας.

Καταρχάς, θέλω να ευχαριστήσω τον επιβλέποντα καθηγητή μου για την πολύτιμη καθοδήγηση και τις εξαιρετικές συμβουλές του καθ' όλη τη διάρκεια της εκπόνησης της. Η εμπειρία και η εμπιστοσύνη του με έκαναν να νιώσω ασφαλής στην ανάπτυξη και υλοποίηση της.

Επίσης, δεν θα μπορούσα να παραλείψω να ευχαριστήσω τους συμφοιτητές μου και τους φίλους μου για τη συνεχή υποστήριξη και την αμοιβαία ενέργεια που μοιραστήκαμε κατά τη διάρκεια αυτής της πορείας. Οι συζητήσεις και οι συνεισφορές σας αποτέλεσαν πηγή έμπνευσης και εμπλούτισαν το έργο μου.

Τέλος, απευθύνω τις θερμότερες μου ευχαριστίες προς την οικογένειά μου, που υποστήριξε τις προσπάθειές μου με αγάπη και κατανόηση. Η στήριξή τους αποτέλεσε τον βασικό πυλώνα της επιτυχίας μου.

Σας ευχαριστώ όλους για τη συνεισφορά σας σε αυτήν την εργασία και για την υποστήριξη που μου παρείχατε σε αυτήν την σημαντική πορεία της εκπαίδευσής μου.

## Πίνακας περιεχομένων

Περίληψη.....	2
Abstract.....	3
Ευχαριστίες.....	5
Κατάλογος Εικόνων .....	8
Εισαγωγή.....	9
Σκοπός της Εργασίας.....	9
Δομή Εργασίας.....	10
Θεωρητική Επισκόπηση.....	10
Βασικές έννοιες του προτύπου μοντελοποίησης BPMN .....	10
Ορισμός BPMN .....	10
Διαγράμματα Επιχειρησιακής Διαδικασίας BPMN.....	12
Γραφική Σημειογραφία της BPMN.....	13
Διαγράμματα Επιχειρησιακής Διαδικασίας BPMN.....	<b>Error! Bookmark not defined.</b>
Γραφική Σημειογραφία της BPMN.....	<b>Error! Bookmark not defined.</b>
Αντικείμενα Ροής .....	13
Αντικείμενα Σύνδεσης .....	14
Διάδρομοι.....	15
Τεχνουργήματα.....	15
BPM adoption.....	17
Η περίπτωση της AMAZON .....	18
Η περίπτωση της WALMART .....	19
Πλατφόρμες Μοντελοποίησης .....	19
IBM Business Process Manager (IBM BPM): .....	20
Pegasystems Pega Platform: .....	20
Appian Business Process Management Suite (Appian BPM): .....	20
Oracle Business Process Management Suite (Oracle BPM): .....	20
Camunda BPM .....	21
Camunda BPM.....	21
Camunda 7.....	22
Camunda 8.....	27
Πλεονεκτήματα και Μειονεκτήματα .....	31
Εγκατάσταση του Camunda .....	32
Επεξήγηση της self-managed προσέγγισης στο Docker .....	32
Οδηγίες εγκατάστασης Camunda 7 σε Docker.....	33
Υλοποίηση εφαρμογών.....	36

FrontEnd.....	36
BackEnd.....	38
Αποτελέσματα και αξιολόγηση.....	56
Αξιολόγηση αποτελεσμάτων.....	57
Επόμενα βήματα.....	58
Εγκατάσταση των εφαρμογών σε Kubernetes.....	59
CI/CD.....	60
Βιβλιογραφία.....	66



## Κατάλογος Εικόνων

Εικόνα 1 Απεικόνιση Αντικειμένων Ροής .....	13
Εικόνα 2 Απεικόνιση Αντικειμένων Σύνδεσης .....	14
Εικόνα 3 Παράδειγμα Απεικόνισης Διαδρομών .....	15
Εικόνα 4 Παράδειγμα Απεικόνισης Τεχνουργημάτων .....	16
Εικόνα 5 Απεικόνιση των Βασικών Συνιστώσων (component-based model) .....	22
Εικόνα 6 Απεικόνιση Cockpit Portal .....	24
Εικόνα 7 Απεικόνιση Camunda Cockpit Processes .....	24
Εικόνα 8 Απεικόνιση Camunda Cockpit Decision Definition deployed .....	25
Εικόνα 9 Απεικόνιση Camunda Tasklist .....	25
Εικόνα 10 Απεικόνιση Camunda Admin .....	26
Εικόνα 11 Παράδειγμα Απεικόνισης Τροποποιήσεων Camunda .....	27
Εικόνα 12 Απεικόνιση Camunda Operations .....	28
Εικόνα 13 Απεικόνιση Camunda Task list .....	29
Εικόνα 14 Απεικόνιση Ανάλυσης Optimize .....	29
Εικόνα 15 Απεικόνιση Keycloak .....	30
Εικόνα 16 Απεικόνιση Camunda Web Modeler .....	30
Εικόνα 17 Απεικόνιση Επιτυχούς Δημιουργίας Docker δοχείου .....	35
Εικόνα 18 Απεικόνιση Οθόνης Σύνδεσης στο Camunda .....	35
Εικόνα 19 Απεικόνιση Ψηφιακών Υπηρεσιών .....	37
Εικόνα 20 Απεικόνιση Λίστας Ολοκληρωμένων και Ενεργών Αιτήσεων .....	38
Εικόνα 21 Απεικόνιση Οθόνης Κώδικα με "Business Key" .....	43
Εικόνα 22 Απεικόνιση Οθόνης Κώδικα με "Συγκεκριμένο Χρήστη" .....	44
Εικόνα 23 Απεικόνιση Οθόνης Κώδικα με "Ομάδα Υποψηφίων" .....	45
Εικόνα 24 Απεικόνιση Οθόνης Κώδικα με "Ορισμό Διαδικασιών" .....	45
Εικόνα 25 Απεικόνιση Οθόνης Κώδικα με τον "Ορισμό διαδικασιών μιας συγκεκριμένης ομάδας" .....	46
Εικόνα 26 Απεικόνιση Οθόνης Κώδικα με "Ενεργούς Ορισμούς Διαδικασίας για Συγκεκριμένη Ομάδα" .....	46
Εικόνα 27 Απεικόνιση Οθόνης Κώδικα με "Ανενεργούς Ορισμούς Διαδικασίας για Συγκεκριμένη Ομάδα" .....	47
Εικόνα 28 Απεικόνιση Οθόνης Κώδικα με το "Κλειδί της φόρμας για μια Συγκεκριμένη Εργασία" .....	47
Εικόνα 29 Απεικόνιση Οθόνης Κώδικα με "Διαδικασίες για συγκεκριμένο χρήστη" .....	48
Εικόνα 30 Απεικόνιση Οθόνης Κώδικα με "Ενεργές Διαδικασίες για συγκεκριμένο χρήστη" .....	48
Εικόνα 31 Απεικόνιση Οθόνης Κώδικα με "Μεταβλητές μια εργασίας" .....	49
Εικόνα 32 Απεικόνιση Οθόνης Κώδικα με "Ολοκληρωμένες Μεταβλητές" .....	49
Εικόνα 33 Απεικόνιση Οθόνης Κώδικα με "Ενεργές μεταβλητές" .....	50
Εικόνα 34 Απεικόνιση Οθόνης Κώδικα "Suspend Process" .....	50
Εικόνα 35 Απεικόνιση Οθόνης Κώδικα "Activate Process" .....	51
Εικόνα 36 Απεικόνιση Οθόνης Κώδικα "Claim Task" .....	51
Εικόνα 37 Απεικόνιση Οθόνης Κώδικα "Unclaim Task" .....	52
Εικόνα 38 Απεικόνιση Οθόνης Κώδικα "Complete Task" .....	52
Εικόνα 39 Απεικόνιση Οθόνης Κώδικα "Start Process" .....	53
Εικόνα 40 Απεικόνιση Οθόνης Κώδικα "Get Groups" .....	54
Εικόνα 41 Απεικόνιση Οθόνης Κώδικα "Post Groups" .....	55
Εικόνα 42 Απεικόνιση Οθόνης CI/CD .....	61

## Εισαγωγή

### Σκοπός της Εργασίας

Οι ραγδαίες τεχνολογικές εξελίξεις, οι οικονομικές συνθήκες και η παγκοσμιοποίηση έχουν συμβάλει στη διαμόρφωση μια νέας επιχειρηματικής πραγματικότητας, η οποία απαιτεί αναδιαμόρφωση στον τρόπο λειτουργίας της εν γένει. Τα ζητήματα που προκύπτουν από την αδυναμία της προσαρμογής τόσο των επιχειρήσεων, όσο και των δημόσιων φορέων είναι ευρέως γνωστά. Μέσα σε αυτό το πλαίσιο, ο ανασχεδιασμός και η αυτοματισμός των διαδικασιών που διεξάγονται μέσα σε μια επιχείρηση ή σε ένα φορέα γίνονται πρωταρχικοί στόχοι.

Στην παρούσα εργασία, αναλύεται η μοντελοποίηση και ο αυτοματισμός των διαδικασιών μέσω του της πλατφόρμας Camunda, ένα εργαλείο που βασίζεται στο πρότυπο Business Process Model and Notation. Λαμβάνεται ως περίπτωση μελέτης η εφαρμογή του Camunda στον δημόσιο τομέα για την ενδεδειγμένη ανάλυση του τρόπου βελτίωσης των διαδικασιών που διεξάγονται σε ένα κύκλο εργασιών του. Για παράδειγμα, οι δημόσιοι φορείς αντιμετωπίζουν συχνά προβλήματα στην επεξεργασία των αιτημάτων που υποβάλλονται από τους πολίτες. Αυτό συμβαίνει είτε από τον τρόπο που υποβάλλονται τα αιτήματα από πλευράς πολιτών, είτε από τις συνεχείς αλλαγές των νόμων και των κανονισμών, και οι οποίες προστάζουν την προσαρμογή των πληροφοριακών συστημάτων των δημόσιων φορέων σε εκείνες. Αποτέλεσμα αυτού είναι, αφενός, ο πολίτης να εμφανίζεται δυσαρεστημένος, επειδή το αίτημα δεν έχει επεξεργαστεί σε εύλογο χρονικό διάστημα και ο εργαζόμενος να έχει ξοδέψει αντιπαραγωγικές εργατοώρες.

Με το Camunda, οι εργαζόμενοι μπορούν να τροποποιήσουν τις διαδικασίες με μια απλή και εύκολη διαδικασία «drag-and-drop», χωρίς την άμεση ανάγκη για επαφή και επικοινωνία με τους μηχανικούς πληροφορικής. Έτσι, εάν νόμος απαιτεί αλλαγές στα βήματα μιας οποιαδήποτε αίτησης, ο αρμόδιος υπάλληλος μπορεί να τον εφαρμόσει ευθύς αμέσως. Αυτή η ευελιξία μειώνει τον χρόνο που απαιτείται για την επεξεργασία των αιτήσεων. Βέβαια, και στις δυο περιπτώσεις, η συνεργασία των μηχανικών πληροφορικής, που θα εξελίξουν τον κώδικα των συστημάτων, και των υπάλληλων που εργάζονται στον φορέα, κρίνεται απαραίτητη.

Η χρήση του συγκεκριμένου εργαλείου μπορεί να βοηθήσει στην αυτοματοποίηση και απλοποίηση των διαδικασιών, μειώνοντας όσο περισσότερο γίνεται την ανθρώπινη παρέμβαση, και βελτιώνοντας την αποτελεσματικότητα και την ακρίβεια της διαχείρισης αιτήσεων, με αποτέλεσμα την εξοικονόμηση πόρων χρόνου για τον δημόσιο φορέα και τη βελτίωση της εμπειρίας των πολιτών.

## Δομή Εργασίας

Στο πρώτο κεφάλαιο της εργασίας γίνεται θεωρητική επισκόπηση της διαχείρισης των διαδικασιών και πώς το σύστημα της αυτοματοποίησης εκείνων, μέσω συγκεκριμένου εργαλείου όπως είναι το πρότυπο BPMN, μεριμνά στη βελτιστοποίησή τους. Επίσης, παρουσιάζονται οι πλατφόρμες BPMN και τα ιδιαίτερα χαρακτηριστικά τους για κάθε μια ξεχωριστά.

Στο δεύτερο κεφάλαιο αναλύονται διεξοδικά η πλατφόρμα Camunda BPM, οι νέες εκδόσεις, καθώς και τα πλεονεκτήματα και τα μειονεκτήματα της χρήσης της.

Στο τρίτο και τέταρτο κεφάλαιο, παρουσιάζονται η εγκατάσταση και η εφαρμογή των δυο εκδόσεων Camunda σε συγκεκριμένο διακομιστή (on-premises), και ο τρόπος ενσωμάτωσης του με τις τεχνολογίες frontend και backend.

Στο πέμπτο κεφάλαιο αξιολογούνται τα αποτελέσματα μετά την εφαρμογή του Camunda στο σύστημα των δημόσιων φορέων, και πως αξιοποιούνται για την αύξηση της αποδοτικότητας εκείνων.

Στο έκτο κεφάλαιο, αξιολογούνται τα αποτελέσματα της χρήσης του ως ένα καινοτόμο εργαλείο για την αναβάθμιση του δημόσιου φορέα, και τα μελλοντικά πλάνα για περαιτέρω βελτίωση του συστήματος εκτός από τον επιδιωκόμενο σκοπό της συγκεκριμένης εργασίας.

## Θεωρητική Επισκόπηση

### Βασικές έννοιες του προτύπου μοντελοποίησης BPMN

#### Ορισμός BPMN

Η BPMN είναι ένα σύστημα γραφικής μοντελοποίησης που παρακολουθεί και καταγράφει τον κύκλο εργασιών της επιχείρησης, το οποίο με τη σειρά του, μέσω των συμβόλων και σχημάτων, αποτυπώνεται στα Διαγράμματα Επιχειρησιακής Διαδικασίας (BPD). Στόχος είναι, η ανάπτυξη μιας μοντελοποίησης απόλυτα κατανοητής και εφαρμόσιμης, «από όλους τους επιχειρηματικούς χρήστες, από τους επιχειρηματικούς αναλυτές που δημιουργούν τα αρχικά σχέδια των διαδικασιών, στους τεχνικούς προγραμματιστές που είναι υπεύθυνοι για την εφαρμογή της τεχνολογίας που θα εκτελέσει αυτές τις διαδικασίες, και τέλος, στους επιχειρηματίες που θα διαχειρίζονται και θα παρακολουθούν αυτές τις διαδικασίες, όχι μόνο από τους ιδιοκτήτες και τους εργαζομένους μιας επιχείρησης, αλλά και από τους αναλυτές των διεργασιών, έως τους μηχανικούς πληροφορικής που είναι υπεύθυνοι για την υλοποίηση της μοντελοποίησης των διεργασιών»(von Rosing, et all ,2015).

Ένας άλλος ορισμός που αποδίδεται από τον Επαγγελματικό Σύλλογο Διαχείρισης Επιχειρησιακών Διαδικασιών είναι, η «επιστημονική προσέγγιση για τον προσδιορισμό, το σχεδιασμό, την εκτέλεση, την τεκμηρίωση, τη μέτρηση, την παρακολούθηση και τον έλεγχο, τόσο των αυτοματοποιημένων, όσο και των μη αυτοματοποιημένων επιχειρησιακών διαδικασιών για την

επίτευξη συνεπών, στοχευμένων αποτελεσμάτων που ευθυγραμμίζονται με τους στρατηγικούς στόχους ενός οργανισμού»

Η BPMN, σύμφωνα με αρκετές μελέτες, είναι περισσότερη προτιμητέα από σχεδόν όλους τους επαγγελματικούς κλάδους, και αυτό το καθιστά ως ένα «de facto» πρότυπο- εργαλείο στην αυτοματοποίηση των διαδικασιών(Recker J., 2009). Αξίζει να σημειωθεί ότι τα καινοτόμα χαρακτηριστικά του έχουν οδηγήσει στην ανάπτυξη πολλών εργαλείων υποστήριξης, όπως το Activiti , το jBPM(Rademakers, T., 2012) και το Oracle BPM (Das, M., Deb, M., Wilkins, M., 2011).

Η λογική αλληλουχία της λειτουργίας της BPMN βασίζεται στα παρακάτω βήματα:

1. Ανάλυση Επιχειρησιακών Διαδικασιών.
2. Σχεδιασμός & Μοντελοποίηση Επιχειρησιακών Διαδικασιών.
3. Αυτοματοποίηση Επιχειρησιακών Διαδικασιών.
4. Παρακολούθηση Επιχειρησιακών Διαδικασιών.
5. Βελτιστοποίηση Επιχειρησιακών Διαδικασιών.

Ως εφελτήριο για τη υλοποίηση της BPMN είναι ανάλυση των επιχειρησιακών διαδικασιών(van der Aalst, 2016). Η ανάλυση των διαδικασιών εντοπίζει τα σημεία αδυναμίας, και τις ανεπάρκειες του υπάρχοντος συστήματος λειτουργίας. Για αυτό το σκοπό, περιλαμβάνονται συνεντεύξεις απ' όλα τα εμπλεκόμενα μέρη της επιχείρησης, συλλεγεί τα απαιτούμενα στοιχεία και χαρτογραφεί τις μελλοντικές διαδικασίες. Δια μέσου των συνεντεύξεων παρέχονται, ουσιαστικά, τις πληροφορίες για την οργανωτική δομή της επιχείρησης. Εκ του αποτελέσματος, δημιουργούνται λίστες στρατηγικών στόχων, επιχειρηματικών διαδικασιών, δομή επιχείρησης και τέλος, «σχέδιο συνεντεύξεων με τη διοίκηση σε επιχειρησιακό επίπεδο» (Damij, N. et all 2008).

Μόλις καταγραφούν και αναλυθούν οι δυσλειτουργίες του υπάρχοντος συστήματος, ακολουθεί ο εκ νέου σχεδιασμός και η μοντελοποίηση του τρόπου λειτουργίας της επιχείρησης(Silver, Pyke, & Peterson, 2009; Recker, et al., 2015). Σε αυτό το στάδιο, δηλαδή, πραγματοποιείται η συνεχής καταγραφή του κύκλου εργασιών όλων των τμημάτων που την συγκροτούν. Τα εμπλεκόμενα μέρη,μέσω του σχεδιασμού και της μοντελοποίησης, κατανοούν με απόλυτη ακρίβεια τις εργασίες που έχουν να φέρουν εις πέρας(Dumas, et al., 2013; Recker, et al., 2015). Αξιοσημείωτο είναι το γεγονός ότι, η μοντελοποίηση, σύμφωνα με το BPMN έχει σχεδιαστεί για να υποστηρίζει ποικίλους τύπους διαδικασιών και επιτρέπει τη δημιουργία διαδικασιών από άκρο σε άκρο.

Στην συνέχεια, μέσω της BPMN, οι διαδικασίες εντός του κύκλου εργασιών αυτοματοποιούνται. Περιλαμβάνει «την ρύθμιση των ροών εργασίας, τον καθορισμό των επιχειρηματικών κανόνων και την ενσωμάτωσή τους στη ήδη υπάρχοντα πληροφοριακά συστήματα» (Jeston & Nelis, 2014). Η αυτοματοποίηση αποτελεί υψίστης σημασίας στάδιο επειδή εξασφαλίζει ότι οι επιχειρησιακές διαδικασίες πραγματοποιούνται με συνέπεια, «μειώνοντας την εξάρτηση από χειρωνακτικές εργασίες» (Peters, 2018). Επομένως, μειώνει όχι μόνο την ανθρώπινη προσπάθεια αλλά και τα σφάλματα, οδηγώντας σε βελτιωμένη εκτέλεση της διαδικασίας(Jeston & Nelis, 2014; Davenport & Harris, 2007). Ακόμη, η παρακολούθηση των επιχειρησιακών διαδικασιών απεικονίζει σε πραγματικό χρόνο την απόδοση των διαδικασιών που βρίσκονται σε εξέλιξη. Σύμφωνα με τους Davenport & Harris, 2007, «η Διαχείριση Επιχειρησιακών Διαδικασιών προσφέρει πολύτιμες πληροφορίες που μπορούν να καθοδηγήσουν τους λήπτες αποφάσεων σε όλα τα επίπεδα».

Η επιχείρηση παρακολουθεί συνεχώς την πρόοδο, καθώς και την απόδοση των αυτοματοποιημένων διαδικασιών χρησιμοποιώντας δείκτες (KPI) και πίνακες ελέγχου (Hasselbring & Shakeri, 2019). Έτσι εντοπίζονται γρήγορα τα σημεία απόκλισης, δίνοντας την ευκαιρία για βελτίωση του συστήματος (Smith & Fingar, 2003). Χάρη σε αυτό, διασφαλίζεται ότι οι διαδικασίες εκτελούνται σύμφωνα με τους κανόνες που έχει ορίσει η BPMN, οπότε και διατηρούν την συνοχή τους με τους στόχους της επιχείρησης.

Κατά το τελευταίο στάδιο, η επιχείρηση αναλύει τα ως άνω για να βελτιστοποιήσει το σύστημα της (Smith & Fingar, 2003). Η βελτιωμένη λήψη αποφάσεων οδηγεί σε πιο αποτελεσματική κατανομή πόρων, διαχείριση των κινδύνων και στρατηγικό σχεδιασμό. Η εταιρεία ανταποκρίνεται άμεσα στις μεταβαλλόμενες συνθήκες αγοράς και στις προτιμήσεις των πελατών. Είναι βέβαιο ότι, αυξάνοντας την αποτελεσματικότητα, αυξάνεται η ποιότητα, οπότε και η ικανοποίηση των πελατών. (Davenport, 2006).

Χρήζει ιδιαίτερης μνείας το γεγονός, η προσέγγιση της BPMN είναι κατεξοχήν πελατοκεντρική. Σύμφωνα με τους Schroeder, et al., 2011; Nasiri, Pishvae, & Rad (2019), με τον συγχρονισμό των διαδικασιών σύμφωνα τις ανάγκες και τις προσδοκίες των πελατών, η επιχείρηση ενισχύει τη συνολική εμπειρία των πελατών.

Αποδεικνύεται από τα παραπάνω ότι, η BPMN προωθεί μια κουλτούρα συνεχούς βελτίωσης, όπου η εταιρεία δεσμεύεται να βελτιώνει συνεχώς τις διαδικασίες της, με στόχο την απόδοση της, και απώτερο σκοπό την αύξηση των κερδών της (Dumas, et al., 2013).

#### Διαγράμματα Επιχειρησιακής Διαδικασίας BPMN

Η BPMN είναι ένα σύνολο από χαρακτηριστικά και προϋποθέσεις, τα οποία καθορίζουν το Διάγραμμα Επιχειρησιακής Διαδικασίας (Business Process Diagram). Το εν λόγω διάγραμμα αναπαριστά με γραφικό τρόπο, δηλαδή μέσω σχημάτων, ένα κύκλο ροής επιχειρησιακών διαδικασιών. Τα σύμβολα αναπαριστούν, όχι μόνο τις δραστηριότητες, αλλά και τα σημεία ελέγχου μιας διαδικασίας σε ένα κύκλο ροής.

Βασική αρμοδιότητα των σχημάτων είναι οι εργαζόμενοι της επιχείρησης να κατανοούν με ακρίβεια τι θέλουν. Με αυτό τον τρόπο, προάγεται η συνεργασία μεταξύ των εργαζομένων που εμπλέκονται σε μια διαδικασία, την ίδια στιγμή που η πολυπλοκότητα της διαδικασίας συνεχίζει να υφίσταται.

Ο συμβολισμός του BPMN μπορεί να εφαρμοστεί σε πολλά είδη διαδικασιών. Το σχέδιο του Business Process Modeling and Notation (BPMN) έχει δημιουργηθεί, έτσι ώστε να μπορεί να καλύπτει μια ευρεία γκάμα περιπτώσεων χρήσης, και να επιτρέπει την ολοκληρωμένη μοντελοποίηση επιχειρηματικών διαδικασιών.

Συνήθως, η μοντελοποίηση χρησιμοποιείται για την ανταλλαγή ποικίλων πληροφοριών μεταξύ πολλών ενδιαφερομένων. Σε διάφορες περιπτώσεις, συμμετέχουν ποικίλοι οργανισμοί και εταιρίες που συνεργάζονται μεταξύ τους. Για αυτόν τον λόγο, το BPMN χρησιμοποιείται όχι μόνο για τη μοντελοποίηση εσωτερικών διαδικασιών σε μια ομάδα αλλά και για τη μοντελοποίηση δημόσιων διαδικασιών (White, S. A. 2004).

## Γραφική Σημειογραφία της BPMN

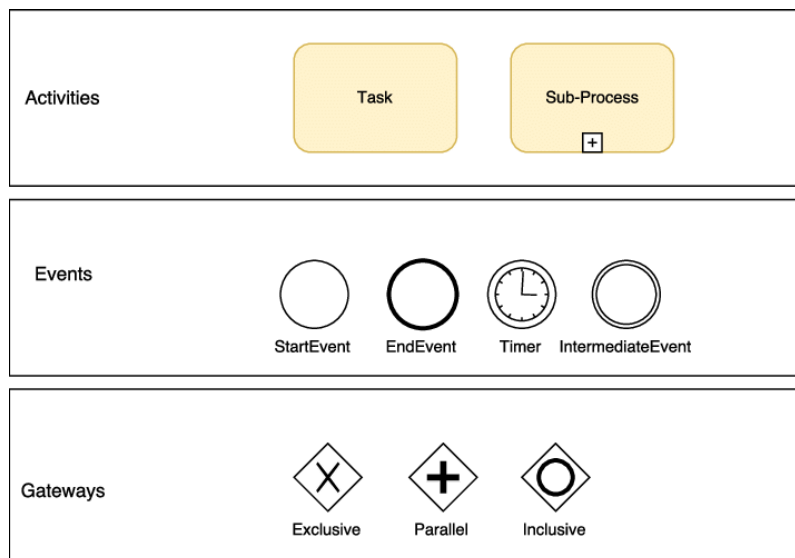
Η πρώτη έκδοση BPMN 1.x προσφέρει μια γραφική σημειογραφία για την πιστή απεικόνιση μιας επιχειρησιακής διαδικασίας σε ένα Διάγραμμα Επιχειρησιακής Διαδικασίας. Σύμφωνα με τον Stephen A. White (2004).

IBM Corporation Η γραφική σημειογραφία της διαθέτει τέσσερις κατηγορίες γραφικών στοιχείων για την κατασκευή διαγραμμάτων.

- τα Αντικείμενα Ροής (Flow Objects),
- τα Αντικείμενα Σύνδεσης (Connecting Objects),
- οι Διάδρομοι (Swimlanes) και
- τα Τεχνουργήματα (Artifacts).

### Αντικείμενα Ροής

Όπως παρατηρείτε+ στην Εικόνα 1, στο πλαίσιο της BPMN, τα Αντικείμενα Ροής ανήκουν σε τρεις κατηγορίες: τα Γεγονότα (Events), τις Δραστηριότητες (Activities) και τις Πύλες (Gateways).



Εικόνα 1 Απεικόνιση Αντικειμένων Ροής

Για να το εξηγήσουμε αναλυτικότερα, τα Γεγονότα σε μια διαδικασία αναπαρίστανται ως κύκλοι και αντιπροσωπεύουν κάτι που συμβαίνει κατά τη διάρκεια της διαδικασίας. Κατά την διαδικασία της μοντελοποίησης, υπάρχουν τρία κύρια είδη γεγονότων: το Γεγονός Έναρξης (Start Event), το Ενδιάμεσο Γεγονός (Intermediate Event) και το Γεγονός Τέλους (End Event) (Owen, M., & Raj, J. 2005).

Οι Δραστηριότητες αναπαρίστανται με παραλληλόγραμμα σχήματα και στρογγυλεμένες γωνίες, και αντιπροσωπεύουν τις ενέργειες που λαμβάνουν χώρα κατά την διάρκεια της διαδικασίας. Οι δραστηριότητες μπορούν να είναι ατομικές (atomic) ή σύνθετες (non-atomic). Κατά την μοντελοποίηση, υπάρχουν δύο κύριες κατηγορίες δραστηριοτήτων: η Εργασία (Task) και η Υπό-διαδικασία (Sub-Process).

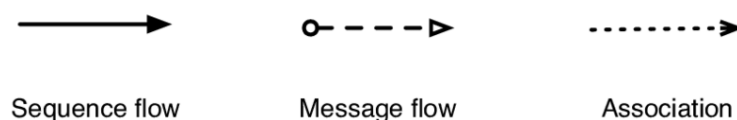
Οι Πύλες αναπαρίστανται με ρόμβους και χρησιμοποιούνται για τον διαχωρισμό και τον συντονισμό των ροών σε ένα διάγραμμα BPMN (συμπεριλαμβανομένων των αποφάσεων για απόκλιση ή σύγκλιση). Κάθε πύλη είναι το σημείο όπου λαμβάνονται αποφάσεις σχετικά με το πώς θα συνεχίσει η ροή μιας διαδικασίας. Ανάλογα με το αποτέλεσμα εξόδου της πύλης, η ροή συνεχίζεται αναλόγως. Επιπλέον, υπάρχουν διάφορα είδη πυλών που μπορούν να χρησιμοποιηθούν ανάλογα με την λογική της διαδικασίας όπως είναι (Owen, M., & Raj, J. 2005):

Αποκλειστικό σύμβολο (Exclusive symbol or XOR).

- Αποκλειστική πύλη βασισμένη σε γεγονότα (Exclusive- Event Based).
- Παράλληλη πύλη (Parallel-AND).
- Πύλη Διάζευξης (Inclusive-OR).
- Σύνθετη πύλη (Complex).
- Πύλη παράλληλων γεγονότων (Parallel Event Based).

#### Αντικείμενα Σύνδεσης

Τα σύμβολα διασύνδεσης αντιπροσωπεύουν γραμμές που έχουν ως βασικό ρόλο τη σύνδεση και την επικοινωνία μεταξύ των αντικειμένων που βρίσκονται σε μια ροή επιχειρηματικής διαδικασίας. Αυτά τα σύμβολα διασύνδεσης χωρίζονται σε τρεις βασικές κατηγορίες:

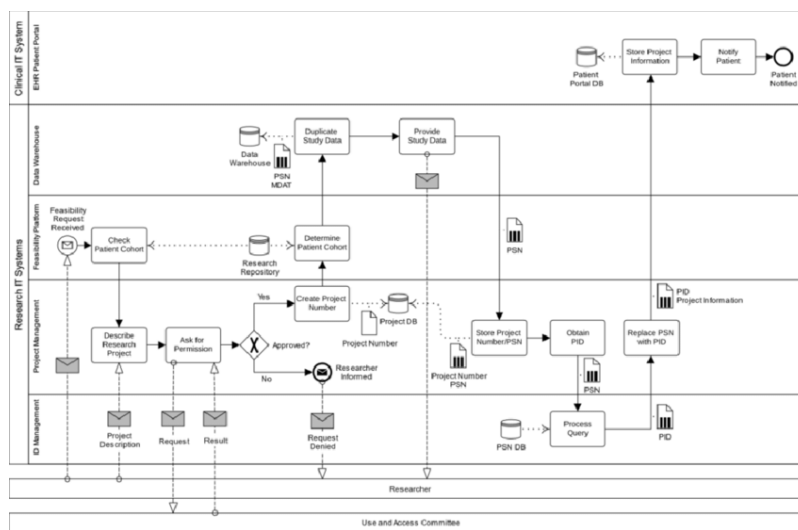


Εικόνα 2 Απεικόνιση Αντικειμένων Σύνδεσης

- Σύμβολο ροής ακολουθίας (Sequence Flow): Αυτό το σύμβολο δείχνει την σειρά και τη συνεχόμενη εκτέλεση των αντικειμένων σε μια επιχειρηματική διαδικασία. Βασικά, αναπαριστά την χρονική σειρά των δραστηριοτήτων και των γεγονότων.
- Σύμβολο ροής μηνυμάτων (Message Flow): Αυτό το σύμβολο χρησιμοποιείται για την ανταλλαγή μηνυμάτων μεταξύ διαφορετικών αντικειμένων σε μια διαδικασία. Συνήθως χρησιμοποιείται για να δείξει την επικοινωνία μεταξύ διαφορετικών διαδικασιών ή διαφορετικών μονάδων εργασίας.
- Σύμβολο συσχέτισης (Association): Αυτό το σύμβολο χρησιμοποιείται για να συνδέσει ένα αντικείμενο με πληροφορίες ή πρόσθετες πληροφορίες που αφορούν σε αυτό, χωρίς να επηρεάζει την ροή της επιχειρηματικής διαδικασίας. Συνήθως χρησιμοποιείται για την παροχή επιπλέον πληροφοριών ή σχολίων πάνω σε ένα αντικείμενο (Analytical Evaluation

## Διάδρομοι

Οι Διάδρομοι είναι μέσα που χρησιμοποιούνται για να οργανώσουν και να καθορίσουν τις ρόλους και τις ευθύνες σε μια επιχειρησιακή διαδικασία. Βασικά, αναπαριστούν τους διάφορους χρήστες ή τα τμήματα που συμμετέχουν στη διαδικασία. Υπάρχουν δύο είδη Διαδρομών: οι Δεξαμενές και οι Λωρίδες. Και τα δύο είδη αντιπροσωπεύουν είτε συγκεκριμένες οντότητες (όπως ένα τμήμα πωλήσεων) είτε γενικούς ρόλους (όπως ένας πωλητής). Οι Λωρίδες αποτελούν υποκατηγορίες των Δεξαμενών και πάντα ανήκουν σε μια Δεξαμενή ή σε μια άλλη Λωρίδα (Lam, V. S. 2009).

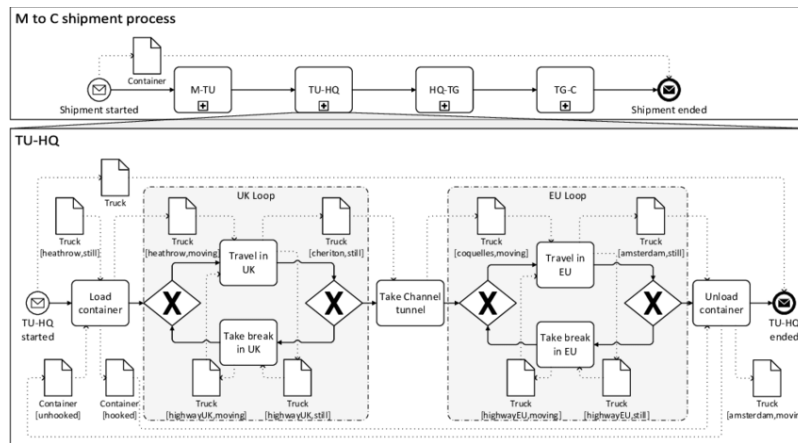


Εικόνα 3 Παράδειγμα Απεικόνισης Διαδρομών

## Τεχνουργήματα

Τα Τεχνουργήματα χρησιμοποιούνται για να παρέχουν επιπλέον πληροφορίες σχετικά με μια διαδικασία, χωρίς να επηρεάζουν τον τρόπο με τον οποίο εκτελείται η διαδικασία. Αυτά τα τεχνουργήματα περιλαμβάνουν τα εξής.





Εικόνα 4 Παράδειγμα Απεικόνισης Τεχνουργημάτων

- Αντικείμενα Δεδομένων (Data Objects): Αντιπροσωπεύουν δεδομένα ή πληροφορίες που συνδέονται με τη διαδικασία. Αυτά τα αντικείμενα δεδομένων δείχνουν τι είδους πληροφορίες χρησιμοποιούνται ή παράγονται κατά την εκτέλεση της διαδικασίας.
- Σημειώσεις (Annotations): Αποτελούν σημειώσεις ή σχόλια που προστίθενται σε ένα διάγραμμα διαδικασίας για να εξηγήσουν περαιτέρω τις λεπτομέρειες ή τον σκοπό της διαδικασίας. Δεν επηρεάζουν τη ροή της διαδικασίας, αλλά προσφέρουν περαιτέρω κατανόηση.
- Ομάδες (Groups): Αναπαριστούν ομάδες αντικειμένων ή διαδικασιών που συνδέονται ή οργανώνονται μαζί σε ένα διάγραμμα διαδικασίας. Οι ομάδες βοηθούν στην οργάνωση του διαγράμματος, αλλά δεν επηρεάζουν τον τρόπο εκτέλεσης των διαδικασιών που περιλαμβάνουν (Lam, V. S. 2009).

Η δεύτερη έκδοση BPMN 2.0 διευρύνει τις δυνατότητες της. Σύμφωνα με τους Michele Chinosi και Alberto Trombetta (Chinosi, M., & Trombetta, A. 2012) επισημοποιεί τις Σημασιολογίες Εκτέλεσης (Execution Semantics) για όλα τα στοιχεία, ορίζει έναν μηχανισμό τόσο τις γραφικές επεκτάσεις όσο και αυτές του μοντέλου διαδικασιών, βελτιώνει τη σύνθεση και συσχέτιση των Γεγονότων, επεκτείνει τον ορισμό των ανθρώπινων αλληλεπιδράσεων, ορίζει τα μοντέλα Χορογραφίας (Choreography) και Συνομιλίας (Conversation), επιλύει γνωστές ασυνέπειες και ασάφειες της BPMN 1.2 και τέλος, ορίζει ένα μεταμοντέλο και ένα μοντέλο ορισμού διαγραμμάτων.

Είναι δυνατή η δημιουργία και η ανάπτυξη τριών διαφορετικών τύπων μοντέλων:

1. Ενορχήστρωση διεργασιών (Processes Orchestration). Αυτό το μοντέλο συμπεριλαμβάνει τρεις τύπους διεργασιών όπως είναι οι ιδιωτικές μη εκτελέσιμες εσωτερικές διεργασίες, οι ιδιωτικές εκτελέσιμες εσωτερικές διεργασίες και οι δημόσιες διεργασίες.
2. Συνεργασίες διεργασιών (Process Collaborations). Αυτό το μοντέλο αφορά τη συνεργασία και τον συντονισμό διαφορετικών διεργασιών.
3. Χορογραφίες διεργασιών (Process Choreographies). Αυτό το μοντέλο αναφέρεται στον συντονισμό των διεργασιών χωρίς να καθορίζει αναλυτικά τον τρόπο με τον οποίο οι διεργασίες λειτουργούν ανεξάρτητα (Allweyer, T. 2016).

## BPM adoption

Είναι ευρέως γνωστό ότι επιχειρήσεις υψηλού κύρους και διεθνούς εμβέλειας επενδύουν πλέον στην εξέλιξη τους, υιοθετώντας τη BPMN. Εντούτοις, οι προκλήσεις που αναδύονται σχετικά είναι πολλές. Για αυτό, κρίνεται αναγκαία η άμεση αντιμετώπιση τους (Richardson, et al, 2015).

Από τις πρώτες προκλήσεις, είναι ο νέος τρόπος λειτουργίας της εταιρείας μετά την εγκατάσταση της ψηφιακής επιχείρησης, και πως εκείνος μπορεί να γίνει εύκολα και γρήγορα αποδεκτός από τα εμπλεκόμενα μέρη (Davenport, 1993). Ο Smith, J. (2020) αναφέρει ότι, «όταν οι διαδικασίες υφίστανται σημαντικό ανασχεδιασμό και αυτοματοποίηση, οι υπάλληλοι μπορεί να νιώθουν ανησυχία για τις αλλαγές». Αποτελεσματικές, επομένως, στρατηγικές διαχείρισης της αλλαγής είναι απαραίτητες για να αντιμετωπιστεί αυτή η πρόκληση, συμπεριλαμβανομένης της σαφούς επικοινωνίας, εκπαίδευσης και συμμετοχής των εμπλεκόμενων μερών στη διαδικασία βελτίωσης (Pritchard, Juttner, & Johnston, 2016). Οι επιχειρήσεις πρέπει να αναδείξουν τα οφέλη της BPMN στους υπαλλήλους και να παρέχουν την απαραίτητη υποστήριξη για να τους βοηθήσουν να προσαρμοστούν σε νέους τρόπους εργασίας. Η αντίσταση στην αλλαγή μπορεί να εμποδίσει την επιτυχή υιοθέτηση των πρακτικών και των τεχνολογιών της BPMN.

Από την άλλη, πρόκληση αποτελεί η προσαρμογή στις ανάγκες των πελατών. Η μοντελοποίηση και η αυτοματοποίηση επικεντρώνονται κυρίως στη βελτίωση της εμπειρίας των πελατών. Επομένως, οι εταιρείες πρέπει να κατανοήσουν τις ανάγκες και τις προτιμήσεις των πελατών τους, και να προσφέρουν ότι εκείνοι χρειάζονται με αποτελεσματικό τρόπο (Westerman, G., Bonnet, D., McAfee, A. 2014).

Μέσα σε αυτό το πλαίσιο, η ασφάλεια και προστασία δεδομένων των πελατών κρίνεται αναγκαία. Η αυτοματοποίηση των διαδικασιών, συνήθως, σημαίνει αύξηση των ψηφιακών δεδομένων που συλλέγονται και αποθηκεύονται από τις επιχειρήσεις. Αυτό με την σειρά του, απαιτεί την εφαρμογή αποτελεσματικών μέτρων ασφαλείας και πολιτικών προστασίας δεδομένων προκειμένου να διασφαλιστεί η εμπιστοσύνη των πελατών, και να αποτραπούν πιθανές παραβιάσεις (McAfee, A., Brynjolfsson, E. 2017).

Επιπλέον, ορισμένες εταιρείες μπορεί να αντιμετωπίζουν τεχνολογικούς περιορισμούς στην υπάρχουσα υποδομή τους ή να αντιμετωπίζουν προβλήματα στην ενσωμάτωση της BPMN. Τα παλαιά πληροφοριακά συστήματα, ο περιορισμένος χώρος αποθήκευσης δεδομένων και οι διαφορετικές τεχνολογίες αποτελούν ανασταλτικούς παράγοντες κατά τη διάρκεια της αυτοματοποίησης των επιχειρησιακών διαδικασιών. Για αυτό, χρειάζεται να γίνουν επενδύσεις στην αναβάθμιση της υποδομής και στην ανάπτυξη τεχνολογικών λύσεων που θα υποστηρίζουν την μοντελοποίηση και την αυτοματοποίηση των επιχειρηματικών διαδικασιών (Ross, J. W., Beath, C., Quaadgras, A. 2013).

Η κυριότητα των διαδικασιών μπορεί να είναι ανασταλτική στην υλοποίηση της BPMN (Pritchard, Juttner, & Johnston, 2016). Υπό την έννοια ότι, «η σαφής ευθύνη είναι απαραίτητη για κάθε διαδικασία προκειμένου να διασφαλιστεί ότι αυτή παρακολουθείται, βελτιώνεται και ευθυγραμμίζεται με τους στόχους της οργάνωσης. Η θέσπιση σαφών ρόλων και ευθυνών για τους κατόχους των διαδικασιών είναι κρίσιμη αποφευχθεί η ασάφεια και για να διασφαλιστεί ότι οι διαδικασίες λαμβάνουν την απαραίτητη προσοχή για τη βελτίωσή τους. Οι κάτοχοι διαδικασιών είναι υπεύθυνοι για την επίβλεψη της απόδοσης των αντίστοιχων διαδικασιών, για τη λήψη

αποφάσεων σχετικά με τις αλλαγές των διαδικασιών, και για τη διασφάλιση ότι οι βελτιώσεις εφαρμόζονται αποτελεσματικά» (Dumas, et al., 2013).

Η μελέτη δύο κορυφαίων εταιρειών όπως είναι η AMAZON και WALMART που χρησιμοποιούν το σύστημα της BPMN μεριμνά στην κατανόηση της λειτουργίας της, τόσο στο εσωτερικό οικοδόμημα τους, όσο και στο εξωτερικό.

### Η περίπτωση της AMAZON

Για την υπεροχή έναντι των υπολοίπων στο χώρο του «e-commerce», η Amazon έχει ενσωματώσει στο οικοδόμημα της τη BPMN, το οποίο αποτελεί βασικό συστατικό για την λειτουργία της (Sarkis, 2020).

Η στρατηγική της Amazon βασίζεται σε ένα μεθοδευμένο δίκτυο αποθηκών βάσει του BPM (He et al., 2020). Στην πράξη, μέσω εκείνου, η εταιρεία έχει επιλέξει τη διατήρηση πολλών αποθηκών που φιλοξενούν υπέρογκες ποσότητες εμπορευμάτων, και ταυτόχρονα, έχουν την δυνατότητα να συντονίζονται και συνεργάζονται όλες μεταξύ τους.

Πιο αναλυτικά, η Amazon, μέσω της BPMN, «παρακολουθεί σε πραγματικό χρόνο τα επίπεδα αποθεμάτων, προβλέπει με ακρίβεια τη ζήτηση και αυτοματοποιεί τη διαδικασία εκπλήρωσης των παραγγελιών. Αυτό επιτρέπει στην εταιρεία να μειώνει τα λάθη, να μειώνει τους χρόνους επεξεργασίας παραγγελιών και να διασφαλίζει τη γρήγορη αποστολή των προϊόντων στους πελάτες».

Παράλληλα, η ανάπτυξη συγκεκριμένων αλγορίθμων BPMN, εξασφαλίζει την πελατοκεντρική προσέγγιση της εταιρείας. Αυτό σημαίνει ότι, υφίσταται ένας «μηχανισμός προτάσεων, ο οποίος προτείνει προϊόντα στους πελάτες με βάση την περιήγησή τους και τις προηγούμενες αγορές τους. Οι αλγόριθμοι αναλύουν τεράστιους όγκους δεδομένων για να προσφέρουν εξατομικευμένες προτάσεις προϊόντων, βελτιώνοντας έτσι την εμπειρία των πελατών και αυξάνοντας τις πωλήσεις» (Chen et al., 2019).

Εκτός από τις λειτουργίες που αφορούν τους πελάτες, η Amazon «βασίζεται επίσης στην BPMN για τη διαχείριση της αλυσίδας εφοδιασμού της. Η αυτοματοποίηση της διαδικασίας επιτρέπει στην εταιρεία να βελτιστοποιεί την μεταφορά των προϊόντων από τους προμηθευτές στα κέντρα διανομής και, τελικά, στις πόρτες των πελατών». Αυτό το επίπεδο βελτιστοποίησης, μειώνει όχι μόνο τα κόστη, αλλά υποστηρίζει, επίσης, τη δέσμευση της Amazon για γρήγορες και αξιόπιστες παραδόσεις (Nasiri et al., 2019).

Παράλληλα, η εταιρεία παρέχει πλατφόρμες υπολογιστικού νέφους, τις Amazon Web Services, οι οποίες χρησιμοποιούν τη BPMN για τη διαχείριση και την επέκταση εκείνων με αποτελεσματικό τρόπο. Παρέχει προηγμένα εργαλεία και καινοτόμες υπηρεσίες BPMN σε αμέτρητες επιχειρήσεις, επιτυγχάνοντας την εύρυθμη λειτουργία τους.

Η στρατηγική προσέγγιση της Amazon, μέσω της BPMN, έχει συμβάλει σημαντικά στην παγκόσμια άνοδο της στον τομέα του e-commerce.

Εν κατακλείδι, η αυτοματοποίηση των διαδικασιών σε τομείς όπως η διαχείριση αποθεμάτων, η εξυπηρέτηση παραγγελιών, οι προσαρμοσμένες προτάσεις και οι υπηρεσίες νέφους, προσφέρει

στην Amazon την δυνατότητα να θέτει νέα πρότυπα αποτελεσματικότητας και ικανοποίησης των πελατών στην ψηφιακή εποχή.

#### Η περίπτωση της WALMART

Η Walmart, η διεθνούς κλίμακας εταιρεία στον λιανισμό, έχει ενσωματώσει στην λειτουργία της το σύστημα της BPMN, βελτιώνοντας την διαχείριση της εφοδιαστικής δραστηριότητας της. Με αυτό τον τρόπο, η επιχείρηση διαθέτει το συγκριτικό πλεονέκτημα στη λιανική πώληση προϊόντων (Peters, 2018).

Με χιλιάδες καταστήματα ανά τον κόσμο και με τεράστια ποικιλία προϊόντων, ο ακριβής έλεγχος των αποθεμάτων αποτελεί ζωτικής σημασίας. Πιο ειδικά, «τα εργαλεία και οι μεθοδολογίες της BPMN επιτρέπουν στη Walmart να παρακολουθεί προσεκτικά τα επίπεδα αποθέματος, να αναλύει ιστορικά δεδομένα πωλήσεων και να προβλέπει με ακρίβεια τη ζήτηση». Έτσι, ελαχιστοποιούνται ζητήματα που ανακύπτουν είτε από την υπερβολική αποθήκευση προϊόντων, είτε από την εξάντληση των αποθεμάτων τους (Nasiri et al., 2019).

Επιπλέον, η αυτοματοποίηση της οργάνωσης του ανεφοδιασμού ενισχύει τον ομαλό συντονισμό του μεγάλου δικτύου διανομής και μεταφοράς προϊόντων από τους προμηθευτές στα καταστήματα, ελαττώνοντας βαθμιαία τους χρόνους παράδοσης αυτών (Sarkis, 2020).

Όσον αφορά τα καταστήματά της, η Walmart έχει εφαρμόσει το σύστημα της BPMN για τη ποιότητα των εσωτερικών διαδικασιών τους. Μέσω της αυτοματοποίησης τους, η εταιρεία έχει αυξήσει την παραγωγικότητα των εργαζομένων, έχει εξασφαλίσει ότι τα ράφια των καταστημάτων είναι πάντοτε εφοδιασμένα, και το σημαντικότερο έχει μειώσει τους χρόνους αναμονής των πελατών στα ταμεία. Αυτές οι βελτιώσεις συντελούν στην ενίσχυση της συνολικής εμπειρίας αγορών των πελατών και στην αποτελεσματικότητα της λειτουργίας των καταστημάτων (Peters, 2018).

Επιπλέον, η εν λόγω εταιρεία έχει ενσωματώσει τη BPMN «στις ψηφιακές της πρωτοβουλίες, συμπεριλαμβανομένης της πλατφόρμας e-commerce. Οι αναλύσεις που προκύπτουν, χρησιμοποιούνται για την εξατομίκευση των online εμπειριών αγορών, τη σύσταση προϊόντων και τη βελτιστοποίηση των διαδικασιών λογιστικής για την παράδοση προϊόντων στους online πελάτες της. Η Walmart ανταγωνίζεται αποτελεσματικά τους online λιανοπωλητές και ανταποκρίνεται στις αυξανόμενες απαιτήσεις των ψηφιακών αγοραστών στις λειτουργίες της e-commerce» (Nasiri et al., 2019).

Απόρροια όλων των παραπάνω είναι ότι, αφενός η Walmart στο εσωτερικό της έχει επιτύχει την κάλυψη των αναγκών των πελατών μειώνοντας σημαντικά τα λειτουργικά έξοδα της εταιρείας, και αφετέρου, έχει διακριθεί για την προσαρμοστικότητα της τόσο σε επίπεδο πελατών, όσο και σε επίπεδο συνθηκών αγοράς (Sarkis, 2020).

#### Πλατφόρμες Μοντελοποίησης

### IBM Business Process Manager (IBM BPM):

Το IBM BPM είναι ένα ευρέως αναγνωρισμένο BPM λογισμικό που παρέχει εργαλεία για τη μοντελοποίηση, την αυτοματοποίηση και τη βελτιστοποίηση των επιχειρηματικών διαδικασιών. Για παράδειγμα, η Coca-Cola Enterprises, μία από τις μεγαλύτερες εταιρείες παραγωγής ποτών παγκοσμίως, εφάρμοσε το IBM BPM για να βελτιστοποιήσει τη διαχείριση παραγγελιών της.

Με την αυτοματοποίηση της επεξεργασίας παραγγελιών, τη διαχείριση αποθεμάτων και την παρακολούθηση των παραδόσεων, επιτεύχθηκε βελτιωμένη λειτουργική αποτελεσματικότητα και ικανοποίηση των πελατών (IBM, 2021).

### Pegasystems Pega Platform:

Το Pega Platform είναι μια ολοκληρωμένη λύση BPM που επικεντρώνεται στην αυτοματοποίηση των επιχειρηματικών διαδικασιών με έμφαση στην εξυπηρέτηση των πελατών. Για παράδειγμα, η American Express, μία κορυφαία εταιρεία χρηματοοικονομικών υπηρεσιών, αξιοποίησε το Pega Platform για τη βελτίωση των διαδικασιών εξυπηρέτησης πελατών της.

Με την αυτοματοποίηση της διαχείρισης περιστατικών, την ανίχνευση απάτης και την επίλυση διαφορών, μειώθηκαν σημαντικά οι χρόνοι απόκρισης και παρέχεται μια πιο ομαλή εμπειρία για τους πελάτες (Pegasystems, 2023).

### Appian Business Process Management Suite (Appian BPM):

Το Appian BPM είναι μια λογισμική λύση που προσφέρεται από την Appian Corporation, μια εταιρεία που εξειδικεύεται σε πλατφόρμες αυτοματισμού χαμηλού κώδικα. Το Appian BPM έχει σχεδιαστεί για να βοηθήσει τις οργανώσεις να απλοποιήσουν και να βελτιστοποιήσουν τις επιχειρηματικές τους διαδικασίες παρέχοντας εργαλεία για τη μοντελοποίηση, τον αυτοματισμό και την παρακολούθηση του κύκλου εργασιών των επιχειρηματικών δραστηριοτήτων.

Με το Appian BPM, οι εταιρείες μπορούν να δημιουργήσουν, να εκτελέσουν και να διαχειριστούν τις διαδικασίες τους με πιο αποτελεσματικό τρόπο, να βελτιώσουν τη συνεργασία μεταξύ των ομάδων και να αποκτήσουν εισαγωγικά στοιχεία για την απόδοση των διαδικασιών μέσω αναλύσεων και αναφορών (Appian, 2023).

Πολλές εταιρείες σε διάφορες βιομηχανίες χρησιμοποιούν το Appian BPM για τη βελτίωση των λειτουργιών τους. Ένα παράδειγμα είναι η Barclays, μια παγκόσμια εταιρεία χρηματοοικονομικών υπηρεσιών. Η ίδια έχει εφαρμόσει το Appian BPM για διάφορους σκοπούς, συμπεριλαμβανομένης της διαχείρισης της συμμόρφωσης και των διαδικασιών εγγραφής πελατών. Χρησιμοποιώντας το Appian BPM, η Barclays κατάφερε να βελτιώσει την αποδοτικότητα των διαδικασιών και τη συμμόρφωση, προσφέροντας παράλληλα μια καλύτερη εμπειρία στους πελάτες της.

### Oracle Business Process Management Suite (Oracle BPM):

Το Oracle BPM είναι μια πλατφόρμα που συνδυάζει BPM, κανόνες και διαχείριση περιστατικών για την βελτιστοποίηση των επιχειρηματικών διαδικασιών από άκρο σε άκρο. Για παράδειγμα, η Vodafone, μια παγκόσμια τηλεπικοινωνιακή εταιρεία, χρησιμοποίησε το Oracle BPM για τη βελτίωση της διαδικασίας εντοπισμού πελατών. Με την αυτοματοποίηση της επαλήθευσης των εγγράφων, την παραγωγή συμβολαίων και τις ροές ενεργοποίησης, επιτεύχθηκε γρηγορότερη ένταξη των πελατών και αύξηση της λειτουργικής αποτελεσματικότητας (Oracle, 2023).

## Camunda BPM

Το Camunda είναι μια ανοιχτού κώδικα πλατφόρμα BPMN που παρέχει μια απλή και εύχρηστη λύση για την αυτοματοποίηση των επιχειρησιακών διαδικασιών. Για παράδειγμα, η Zalando, μια γνωστή εταιρεία ηλεκτρονικού εμπορίου, υιοθέτησε το Camunda BPMN για τη βελτιστοποίηση της διαδικασίας διαχείρισης επιστροφών. Με την αυτοματοποίηση των αιτήσεων επιστροφής, των διαδικασιών ελέγχου και των διαδικασιών αποζημίωσης, επιτεύχθηκαν γρηγορότεροι χρόνοι επεξεργασίας, μείωση των δαπανών και βελτίωση της ικανοποίησης των πελατών (Camunda, 2023).

## Camunda BPM

Για την εκπόνηση της εργασίας, έχει επιλεγθεί το λογισμικό της Camunda ως την πιο κατάλληλη πλατφόρμα που θα υποστηρίξει τους δημόσιους φορείς στην βελτιστοποίηση των διαδικασιών τους. Η επιλογή έχει βασιστεί σε διάφορους λόγους που καθιστούν την Camunda ιδανική για το συγκεκριμένο σκοπό, οι οποίοι παρουσιάζονται παρακάτω.

Αρχικά, η Camunda προσφέρει μια πληθώρα εργαλείων και υπηρεσιών που υποστηρίζουν τη διαχείριση των διαδικασιών, καθώς και την αυτοματοποίησή τους. Έτσι, δημιουργούνται εξατομικευμένες εφαρμογές που θα εξυπηρετούν αποτελεσματικά τις ανάγκες των δημόσιων φορέων.

Επίσης, το γεγονός ότι λογισμικό είναι ανοικτού κώδικα, δίνει την δυνατότητα να προσαρμοστεί στις ανάγκες των δημόσιων φορέων.

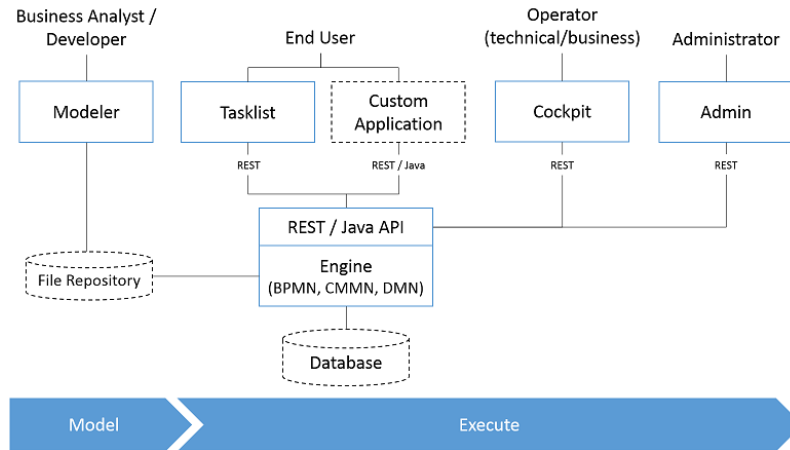
Η Camunda παρέχει προηγμένες δυνατότητες για τη διαχείριση των επιχειρηματικών διαδικασιών, την παρακολούθηση των εργασιών και την ενσωμάτωση με άλλα συστήματα, βοηθώντας τους δημόσιους φορείς να βελτιώσουν την αποτελεσματικότητά τους και να εξοικονομήσουν χρόνο και πόρους.

Διαθέτει δυο βασικές εκδόσεις στην ναυαρχίδα της όπως είναι το Camunda 7 και το Camunda 8. Η πρώτη, που συχνά αναφέρεται ως «Camunda» αποτελεί την προτελευταία έκδοση της πλατφόρμας και είχε κυκλοφορήσει για πρώτη φορά τον Μάρτιο του 2013. Από την άλλη, η Camunda Platform 8 αποτελεί μια περαιτέρω εξέλιξη του «Camunda Cloud», το οποίο βασίζεται στην αυτό-αναπτυσσόμενη μηχανή διαδικασιών Zeebe και διατέθηκε στην αγορά πρώτη φορά τον Απρίλιο του 2022. Τα ιδιαίτερα χαρακτηριστικά των δυο εκδόσεων της Camunda αναλύονται διεξοδικά στο επόμενο κεφάλαιο (Camunda 2023).

## Camunda 7

Η αρχιτεκτονική της πλατφόρμας Camunda εφαρμόζει το μοντέλο των τμημάτων (component-based model). Εκεί κάθε μέρος αναλαμβάνει έναν προσδιορισμένο ρόλο και εκτελεί μια συγκεκριμένη λειτουργία στο σύστημα.

Οι βασικές συνιστώσες (components) περιλαμβάνουν όσα παρουσιάζονται στην παρακάτω εικόνα.



Εικόνα 5 Απεικόνιση των Βασικών Συνιστώσων (component-based model)

- **BPMN Engine:** Ο BPMN engine του Camunda 7 είναι ο πυρήνας που είναι υπεύθυνος για την εκτέλεση των μοντέλων BPMN 2.0. Ερμηνεύει διαγράμματα BPMN, οργανώνοντας εργασίες, γεγονότα και πύλες στις επιχειρηματικές διαδικασίες (BPMN.org, 2021). Αυτός ο πυρήνας είναι ανθεκτικός και ευέλικτος, υποστηρίζοντας λεπτομερή μοντελοποίηση και εκτέλεση διαδικασιών. Στηρίζει ακολουθίες και παράλληλες ροές, βάσει γεγονότων και εργασίες χρηστών, καθιστώντας τον κατάλληλο για διάφορες περιπτώσεις αυτοματοποίησης διαδικασιών (BPMN.org, 2021).
- **DMN Engine:** Το Camunda 7 ενσωματώνει έναν πυρήνα Μοντέλου και Σημειώσεων για τις Απόφασης (DMN), επιτρέποντας στις εταιρείες να λαμβάνουν αποφάσεις βασισμένες στα δεδομένα και στις διαδικασίες τους (OMG, 2017). Αυτός ο πυρήνας εκτελεί πίνακες αποφάσεων και διαγράμματα απαιτήσεων απόφασης (DRD), βελτιώνοντας την ευελιξία της διαδικασίας και επιτρέποντας την λήψη αποφάσεων βασισμένων σε κανόνες (OMG, 2017). Παίζει καθοριστικό ρόλο σε περιπτώσεις όπου οι αποφάσεις καθορίζουν τη ροή της διαδικασίας, όπως η κατεύθυνση των εργασιών ή η καθορισμός της επιλεξιμότητας (OMG, 2017). Ο DMN εξασφαλίζει συνεπείς και ελεγχόμενες αποφάσεις, που είναι κρίσιμες για την αποτελεσματική αυτοματοποίηση των διαδικασιών (OMG, 2017).
- **Tasklist and Cockpit:**  
 Η Λίστα Εργασιών και το Cockpit είναι ζωτικές web εφαρμογές στο Camunda 7. Η Λίστα Εργασιών προσφέρει μια χρήσιμη δι-επαφή για τη διαχείριση και ολοκλήρωση των εργασιών των χρηστών στις διαδικασίες (Camunda, 2023). Το Cockpit, από την άλλη, είναι ένα εργαλείο παρακολούθησης και διαχείρισης, παρέχοντας πραγματικά χρονικά δεδομένα για τις περιπτώσεις των διαδικασιών και την απόδοσή τους (Camunda, 2023). Μαζί, βελτιώνουν την εμπειρία του χρήστη και προσφέρουν βασικά εργαλεία για τη διαχείριση και τη βελτιστοποίηση των διαδικασιών (Camunda, 2023).



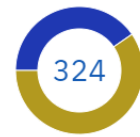
## Right Now



Running Process Instances



Open Incidents



193: Assigned to a user

## Deployed

Process Definitions

9

Decision Definitions

3

Case Definitions

0

Deployments

50

Εικόνα 6 Απεικόνιση Cockpit Portal

Dashboard » Processes » Αίτηση περί μη οφειλής ΤΑΠ : Runtime

Definition Version: 2

Version Tag: null

Definition ID: eservices\_\_tap:2:1215c026-433f-11...

Definition Key: eservices\_\_tap

Definition Name: Αίτηση περί μη οφειλής ΤΑΠ

History Time To Live: 30 days

Tenant ID: null

Deployment ID: 12130104-433f-11ee-a0c0-0242ac...

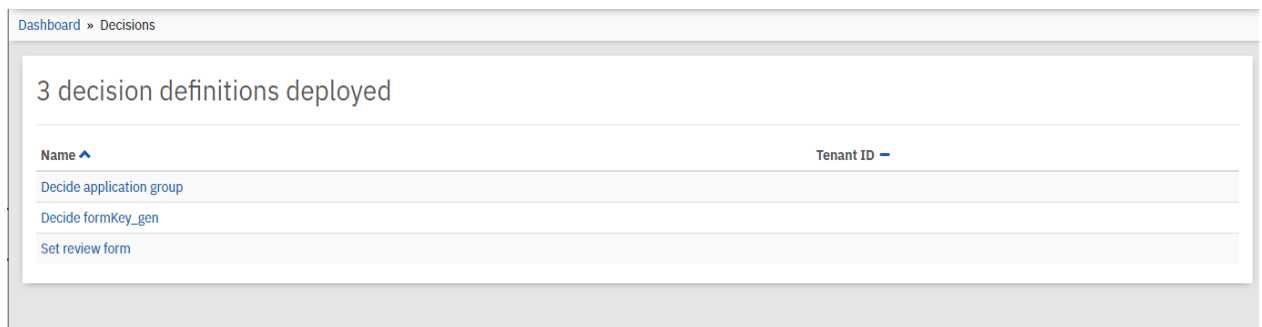
Instances Running:

- current version: 30
- all versions: 30

Process Instances

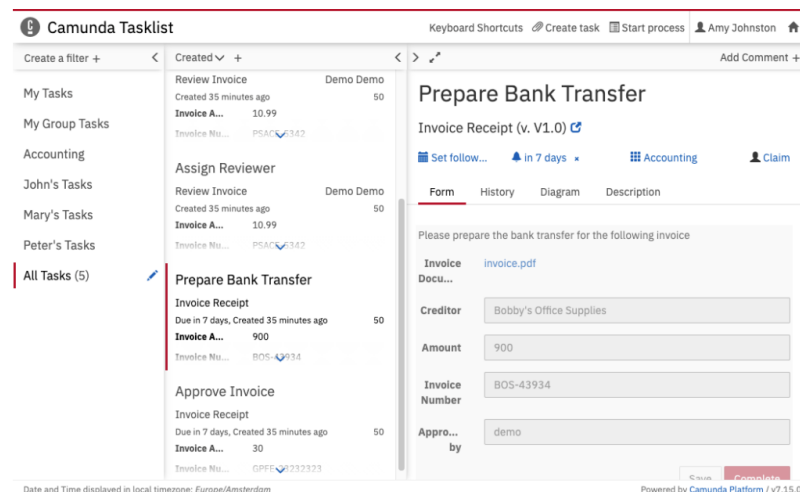
State	ID	Start Time	Business Key
✓	9485733f-6457-11ee-a0c0-0242ac140002	2023-10-06T17:49:41	9G9X1KOH
✓	d8df6a26-6422-11ee-a0c0-0242ac140002	2023-10-06T11:32:12	84U8U637
✓	81e3aa73-6387-11ee-a0c0-0242ac140002	2023-10-05T17:00:14	2X60NAAE
✓	06534e40-636a-11ee-a0c0-0242ac140002	2023-10-05T13:29:11	SJOV8831
✓	620b6d88-62b6-11ee-a0c0-0242ac140002	2023-10-04T16:03:16	6U15FKT1
✓	ca85f159-62a7-11ee-a0c0-0242ac140002	2023-10-04T14:18:49	KL0W92X9

Εικόνα 7 Απεικόνιση Camunda Cockpit Processes



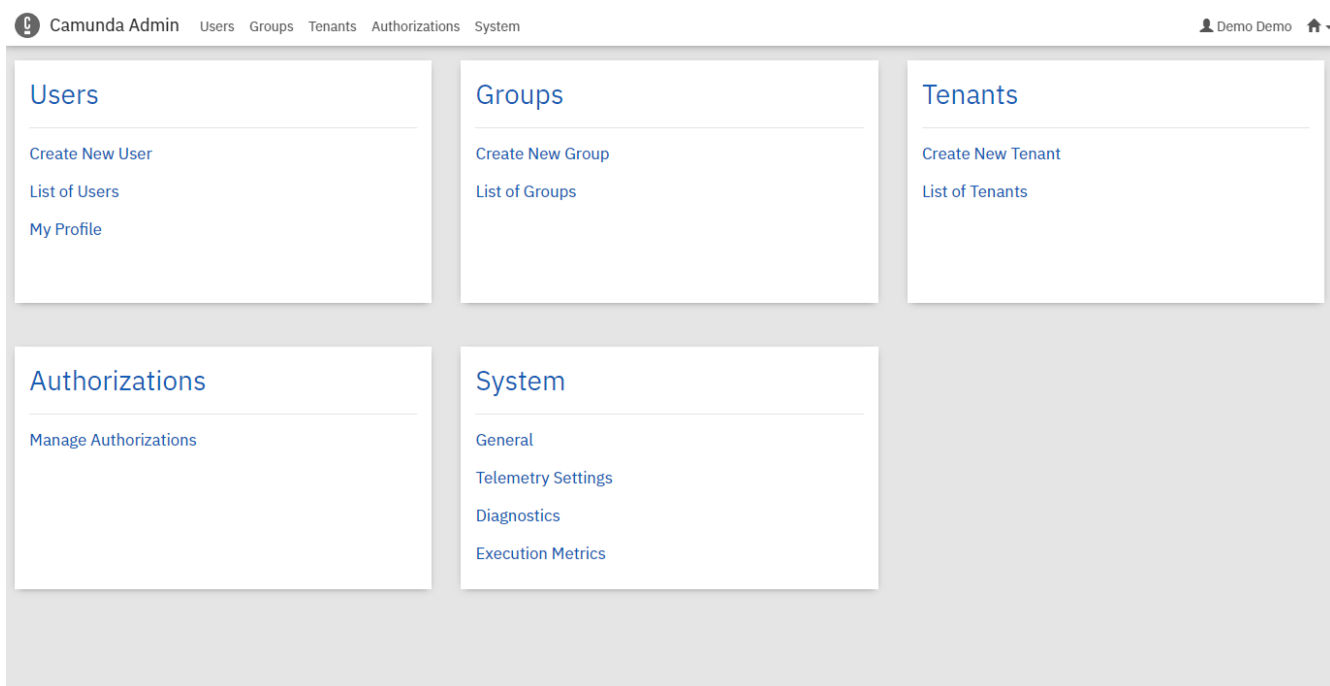
Εικόνα 8 Απεικόνιση Camunda Cockpit Decision Definition deployed

- Modeler:** Ο Μοντελοποιητής του Camunda είναι ένα γραφικό εργαλείο για τη δημιουργία και την επεξεργασία διαγραμμάτων BPMN και DMN (Camunda, 2023). Απλοποιεί τη μοντελοποίηση διαδικασιών, εξασφαλίζοντας ακριβή αναπαράσταση και ευθυγράμμιση με τα πρότυπα BPMN 2.0 και DMN (BPMN.org, 2021). Ο Μοντελοποιητής του Camunda διευκολύνει την αποτελεσματική δημιουργία διαγραμμάτων, ενσωματώνοντας εργασίες, πύλες, γεγονότα και πίνακες αποφάσεων (Camunda, 2023).



Εικόνα 9 Απεικόνιση Camunda Tasklist

- Admin:** Το Camunda Admin προσφέρει πολλές δυνατότητες για τον έλεγχο και τη διαχείριση των χρηστών, των ομάδων, των δικαιωμάτων πρόσβασης και άλλων ρυθμίσεων στο περιβάλλον του Camunda. Με τη χρήση του Camunda Admin, μπορείτε να διαχειριστείτε τους χρήστες και τους ρόλους τους, να δημιουργήσετε και να διαχειριστείτε ομάδες χρηστών, να ορίσετε ποιος έχει πρόσβαση σε τι και να παρακολουθείτε τις δραστηριότητες και τα αρχεία καταγραφής των χρηστών. Επιπλέον, μπορείτε να προσαρμόσετε τον τρόπο πρόσβασης στις διαδικασίες, τα έργα και άλλα στοιχεία του Camunda. Οι δυνατότητες διαχείρισης που προσφέρει το Camunda Admin επιτρέπουν την προσαρμογή του περιβάλλοντος του Camunda BPM για να ικανοποιεί τις ανάγκες του κάθε οργανισμού, παρέχοντας τον απόλυτο έλεγχο και την ευελιξία στη διαχείριση των χρηστών, των ομάδων και άλλων ρυθμίσεων στο περιβάλλον του Camunda BPM.



Εικόνα 10 Απεικόνιση Camunda Admin

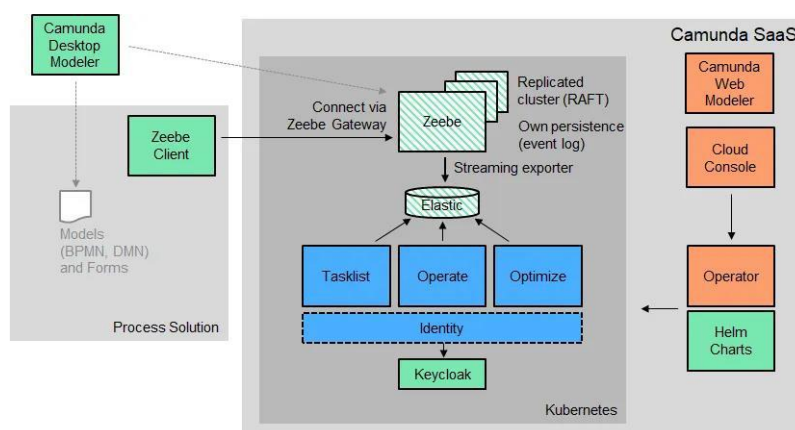
- **REST API and SDKs:** Το Camunda 7 προσφέρει μια REST API και Ανάπτυξη Λογισμικού (SDKs) για διάφορες γλώσσες προγραμματισμού (Camunda, 2023). Αυτά τα εργαλεία διευκολύνουν τη σύνθεση με εξωτερικά συστήματα και υπηρεσίες, επιτρέποντας προγραμματιστική αλληλεπίδραση με το Camunda (Camunda, 2023). Τα REST API και τα SDKs είναι ουσιώδη για την αυτοματοποίηση των διαδικασιών, την ανταλλαγή δεδομένων και τον έλεγχο (Camunda, 2023).
- **Job Executor:** Ο Εκτελεστής Εργασιών στο Camunda 7 είναι κρίσιμος για τη διαχείριση ασύγχρονων εργασιών στις διαδικασίες (Camunda, 2023). Εκτελεί αποτελεσματικά εργασίες που απαιτούν καθυστέρηση, όπως χρονοδιακόπτες και ασύγχρονες δραστηριότητες (Camunda, 2023). Ο Εκτελεστής Εργασιών εξασφαλίζει ανταπόκριση και αδιάλειπτη εκτέλεση των διαδικασιών (Camunda, 2023).
- **External Task Workers:** Το Camunda 7 παρουσιάζει τους Εργαζόμενους Εξωτερικών Εργασιών, επιτρέποντας την εκτέλεση εργασιών από εξωτερικά συστήματα ή υπηρεσίες (Camunda, 2023). Αυτό το χαρακτηριστικό βελτιώνει την ευελιξία και την κλιμάκωση της εκτέλεσης των διαδικασιών, επιτρέποντας τη χρήση υπάρχοντων συστημάτων και υπηρεσιών (Camunda, 2023).
- **Community and Ecosystem:** Το Camunda διαθέτει μια ενεργή κοινότητα ανοικτού κώδικα και ένα οικοσύστημα. Η κοινότητα συνεισφέρει ενεργά στην ανάπτυξη της πλατφόρμας, μοιράζεται γνώσεις και παρέχει υποστήριξη στους χρήστες (Camunda, 2023). Επιπλέον, το οικοσύστημα του Camunda περιλαμβάνει μια αγορά για επεκτάσεις, πρόσθετα και συνδέσεις που βελτιώνουν τις δυνατότητες της πλατφόρμας (Camunda, 2023).

- **Scalability and Clustering:** Το Camunda 7 είναι σχεδιασμένο να υποστηρίζει την κλιμάκωση και την υψηλή διαθεσιμότητα. Υποστηρίζει οριζόντια και κάθετη κλιμάκωση, καθιστώντας το κατάλληλο για την χειρισμό μεγάλων όγκων περιπτώσεων διαδικασίας και εργασιών (Camunda, 2023).
- **Licensing Options:** Το Camunda προσφέρει ένα ευέλικτο μοντέλο αδειας, συμπεριλαμβανομένης μιας κοινοτικής έκδοσης ανοικτού κώδικα και εμπορικών εκδόσεων με επιπρόσθετα χαρακτηριστικά και υποστήριξη (Camunda, 2023). Αυτή η ευελιξία επιτρέπει στις εταιρείες να επιλέγουν την έκδοση που ταιριάζει με τις συγκεκριμένες απαιτήσεις και περιορισμούς τους (Camunda, 2023).

Ανακεφαλαιώνοντας, το Camunda 7 περιλαμβάνει ένα ολοκληρωμένο σύνολο στοιχείων που επιτρέπουν στις εταιρείες να μοντελοποιούν, αυτοματοποιούν και βελτιστοποιούν τις επιχειρηματικές τους διαδικασίες και αποφάσεις. Αυτά τα στοιχεία όπως είναι οι μηχανές και εργαλεία μοντελοποίησης μέχρι REST API και ενσωματώσεις, παρέχουν μια ανθεκτική και ευέλικτη πλατφόρμα για την αυτοματοποίηση διαδικασιών. Η ενεργή κοινότητα του Camunda, και οι επιλογές αδειοδότησης του συμβάλλουν επιπλέον στην έλξη του ως μια ευέλικτη λύση για εταιρείες που επιθυμούν να βελτιώσουν και να απλοποιήσουν τις ροές εργασίας και τη λήψη αποφάσεων.

## Camunda 8

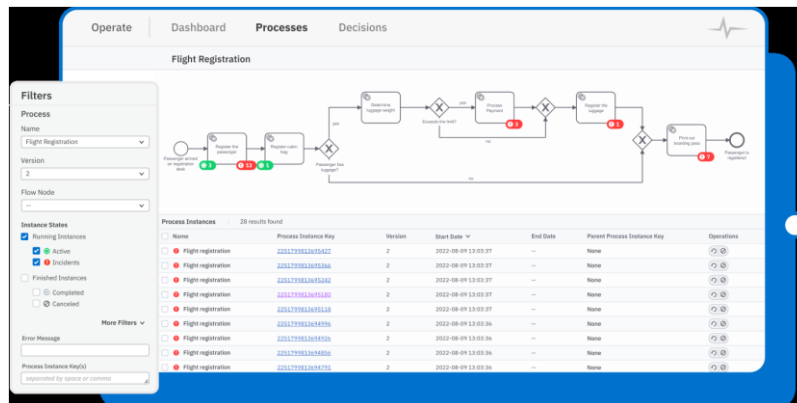
Η αρχιτεκτονική του Camunda 8 έχει υποστεί σημαντικές τροποποιήσεις σε σχέση με το Camunda 7. Μερικές από τις κύριες αλλαγές και στοιχεία στη δομή του Camunda 8 διαφαίνονται στο διάγραμμα της εικόνας παρακάτω:



Εικόνα 11 Παράδειγμα Απεικόνισης Τροποποιήσεων Camunda

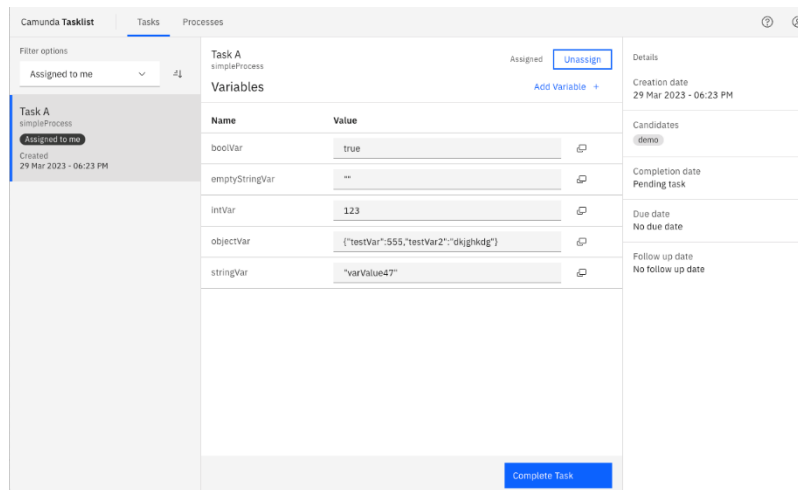
**Zeebe Engine:** Η καρδιά της διαδικασίας στο Camunda 8 είναι το Zeebe Engine, μια διαφορετική μηχανή διαδικασιών από αυτή που χρησιμοποιείται στο Camunda 7. Το Zeebe Engine είναι σχεδιασμένο για να εκτελεί διαδικασίες κλιμάκωσης τόσο οριζόντια όσο και γραμμικά. Είναι κατάλληλο για λειτουργία στον νέφος (cloud) και την αυτοματοποίηση διαδικασιών που περιλαμβάνουν έναν μεγάλο αριθμό διαδικασιών. Επιπλέον, βοηθά στην ανάπτυξη και διαχείριση των ρών εργασίας και τη διαχείριση της κατάστασης κάθε διαδικασίας. Τέλος, παρέχει σημαντικές λειτουργίες για τον σχεδιασμό και τη διαχείριση επιχειρησιακών διαδικασιών, καθώς και τη δυνατότητα αποτελεσματικής διαχείρισης μεγάλης κλίμακας ρών εργασίας που απαιτούν υψηλή διαθεσιμότητα και ανθεκτικότητα σε σφάλματα.

**Operate:** Το Camunda Operate είναι μια εφαρμογή που χρησιμοποιείται για την παρακολούθηση και αναφορά, που παρουσιάζει δεδομένα με οπτικό τρόπο και προσφέρει ανάλυση των διαδικασιών που εκτελούνται. Αναφέρει πληροφορίες και στατιστικά στοιχεία σχετικά με την απόδοση των διαδικασιών και βοηθά στον εντοπισμό πιθανών προβλημάτων.



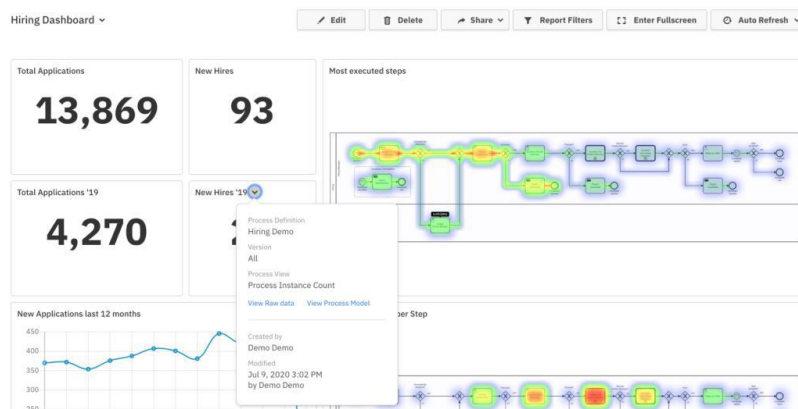
Εικόνα 12 Απεικόνιση Camunda Operations

**Tasklist:** Το Tasklist είναι μια εφαρμογή που παρέχει ένα περιβάλλον όπου οι χρήστες μπορούν να διαχειρίζονται τις εργασίες που σχετίζονται με τις διαδικασίες. Μέσω αυτής της εφαρμογής, οι χρήστες μπορούν να δειναιμικά παρακολουθούν, εκτελούν και διαχειρίζονται τις εργασίες που τους έχουν ανατεθεί.



Εικόνα 13 Απεικόνιση Camunda Task list

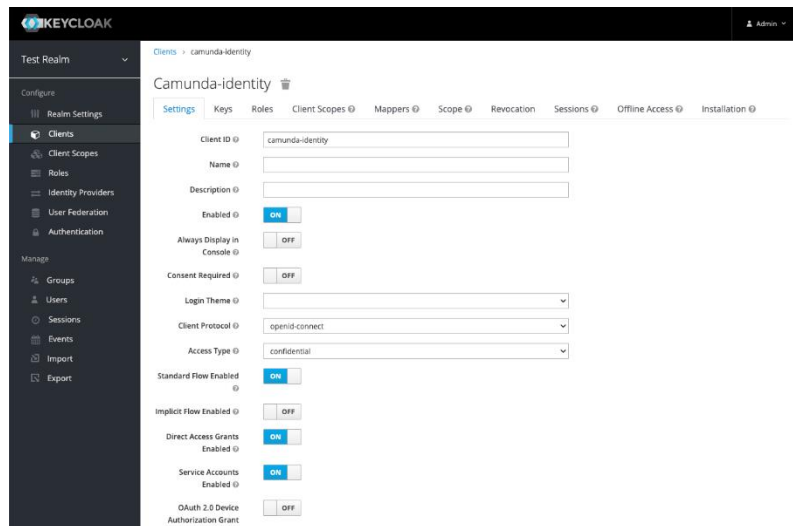
**Optimize:** ο Optimize αποτελεί μια προηγμένη εφαρμογή ανάλυσης και αναφορών που προσφέρει εκτεταμένες δυνατότητες για λεπτομερή αναφορά σχετικά με τις διαδικασίες που εκτελούνται. Παρέχει προηγμένες μετρήσεις, γραφήματα και πίνακες που βοηθούν στην κατανόηση και βελτιστοποίηση των διαδικασιών.



Εικόνα 14 Απεικόνιση Ανάλυσης Optimize

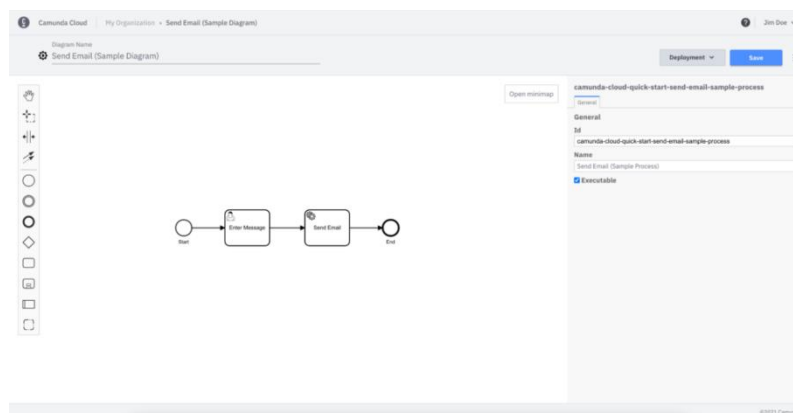
**Identity:** Το Identity είναι μια εφαρμογή που αναλαμβάνει την προστασία και τον έλεγχο της ασφάλειας των χρηστών στο Camunda Platform. Τα καθήκοντά της περιλαμβάνουν την εγγραφή και τη διαχείριση των χρηστών, των ομάδων και των δικαιωμάτων πρόσβασης.

**Keycloak:** Το Keycloak είναι μια πλατφόρμα που λειτουργεί με ανοιχτό κώδικα και ασχολείται με τον έλεγχο ταυτότητας και τη διαχείριση των δικαιωμάτων πρόσβασης. Χρησιμοποιείται για να εξασφαλίσει ότι οι χρήστες που έχουν πρόσβαση στο Camunda Platform είναι αυθεντικοί και έχουν την κατάλληλη άδεια.



Εικόνα 15 Απεικόνιση Keycloak

**Camunda Modeler:** Ο Camunda Web Modeler είναι ένα εργαλείο στο διαδίκτυο που χρησιμοποιείται για να δημιουργήσετε και να επεξεργαστείτε διαγράμματα BPMN, δηλαδή διαγράμματα που απεικονίζουν διαδικασίες εργασίας. Το εργαλείο παρέχει μια φιλική προς τον χρήστη δι-επαφή που σας επιτρέπει να δημιουργήσετε και να επεξεργαστείτε εύκολα αυτά τα διαγράμματα.



Εικόνα 16 Απεικόνιση Camunda Web Modeler

## Πλεονεκτήματα και Μειονεκτήματα

Το εργαλείο Camunda, ως μια αξιόπιστη πλατφόρμα διαχείρισης επιχειρηματικών διεργασιών, προσφέρει πληθώρα πλεονεκτημάτων. Η ευελιξία και η προσαρμοστικότητα της επιτρέπουν στις επιχειρήσεις να μοντελοποιούν και να εκτελούν περίπλοκες ροές εργασιών, προσαρμόζοντάς τη στις ολοένα εξελισσόμενες απαιτήσεις (Dijkman et al., 2011). Ως λύση ανοικτού κώδικα, το Camunda παρέχει μια οικονομικά αποδοτική εναλλακτική, επιτρέποντας στις επιχειρήσεις να διαχειρίζονται τους πόρους τους αποτελεσματικά (Wynn & Eckman, 2017). Με τις εκτεταμένες δυνατότητες ενσωμάτωσης του, το Camunda επιτρέπει ομαλή συνδεσιμότητα με διάφορα εξωτερικά συστήματα και υπηρεσίες, διευκολύνοντας την άριστη ανταλλαγή δεδομένων (Kumari et al., 2020).

Ωστόσο, είναι σημαντικό να σημειωθεί ότι η χρήση του Camunda απαιτεί έναν συγκεκριμένο επίπεδο τεχνικής εμπειρογνομosύνης και εξοικείωσης με τις έννοιες της διαχείρισης επιχειρηματικών διεργασιών (Bala et al., 2019). Οι οργανισμοί μπορεί να χρειαστεί να επενδύσουν σε εκπαίδευση των υπαλλήλων τους ή να προσλάβουν εξειδικευμένο προσωπικό για να διασφαλίσουν την εξειδικευμένη αξιοποίηση και συντήρηση της πλατφόρμας (Yan et al., 2018).

Ως μια πλατφόρμα ανοικτού κώδικα, οι υπηρεσίες υποστήριξης του Camunda μπορεί να είναι περιορισμένες σε σύγκριση με τις εμπορικές λύσεις διαχείρισης επιχειρηματικών διεργασιών. Οι οργανισμοί που εξαρτώνται σημαντικά από ολοκληρωμένες υπηρεσίες υποστήριξης και συντήρησης θα πρέπει να αξιολογήσουν προσεκτικά τις απαιτήσεις υποστήριξής τους και να εξετάσουν επιπλέον επιλογές (Leoni et al., 2020).

Αναλυτικότερα, στα πλεονεκτήματα συγκαταλέγονται η ευελιξία και δυνατότητα προσαρμογής. Το Camunda παρέχει μια ευέλικτη και προσαρμόσιμη πλατφόρμα για τη διαχείριση επιχειρηματικών διεργασιών. Επιτρέπει στις οργανώσεις να μοντελοποιούν και να εκτελούν περίπλοκες ροές εργασιών, να τις προσαρμόζουν σε αλλαγές απαιτήσεων και να τις ενσωματώνουν με διάφορα συστήματα και τεχνολογίες (Dijkman et al., 2011).

Ακόμα, το Camunda είναι μια πλατφόρμα ανοικτού κώδικα, πράγμα που σημαίνει ότι είναι διαθέσιμη ελεύθερα για χρήση και μπορεί να προσαρμοστεί για να ικανοποιήσει τις ειδικές ανάγκες της επιχείρησης. Αυτό μπορεί να οδηγήσει σε εξοικονόμηση κόστους σε σύγκριση με τις κλειστές λύσεις BPM (Wynn & Eckman, 2017).

Επιπλέον, προσφέρει εκτεταμένες δυνατότητες ενσωμάτωσης, επιτρέποντας στις οργανώσεις να συνδεθούν με εξωτερικά συστήματα, βάσεις δεδομένων και υπηρεσίες. Υποστηρίζει διάφορα πρωτόκολλα επικοινωνίας και μορφές δεδομένων, επιτρέποντας ομαλή ενσωμάτωση με υπάρχουσες IT υποδομές (Kumari et al., 2020).

Στα μειονεκτήματα περιλαμβάνονται η μαθησιακή καμπύλη και ειδίκευση. Η χρήση του Camunda απαιτεί ένα συγκεκριμένο επίπεδο τεχνικής εμπειρογνομosύνης και εξοικείωσης με τις έννοιες της διαχείρισης επιχειρηματικών διεργασιών. Οι οργανισμοί μπορεί να χρειαστεί να επενδύσουν σε εκπαίδευση των υπαλλήλων τους ή να προσλάβουν εξειδικευμένο προσωπικό για να χρησιμοποιήσουν και να διατηρήσουν αποτελεσματικά την πλατφόρμα (Mendling et al., 2008).



Επίσης, η έλλειψη χρηστικής διεπαφής: Αν και το Camunda παρέχει ένα ισχυρό σύνολο εργαλείων για το μοντελοποίηση και την εκτέλεση διεργασιών, η δι-επαφή του με τον χρήστη μπορεί να μην είναι τόσο εύκολη στη χρήση όπως σε ορισμένες εμπορικές λύσεις BPM. Αυτό μπορεί να έχει επιπτώσεις στην υιοθέτηση και την παραγωγικότητα των χρηστών (Heuvel & Jansen-Vullers, 2017).

Τέλος, προσφέρει περιορισμένη επιχειρησιακή υποστήριξη. Ως μια πλατφόρμα ανοικτού κώδικα, οι προσφορές υποστήριξης του Camunda μπορεί να είναι περιορισμένες σε σύγκριση με τις εμπορικές λύσεις BPM. Οι οργανισμοί που εξαρτώνται σε μεγάλο βαθμό από πλήρεις υπηρεσίες υποστήριξης και συντήρησης μπορεί να χρειαστεί να αξιολογήσουν τις ανάγκες τους σε υποστήριξη και να εξετάσουν τις επιλογές τους (Zur Muehlen & Indulska, 2010).

## Εγκατάσταση του Camunda

Για την εγκατάστασή του Camunda σε ένα on-premises περιβάλλον και τη δημιουργία ενός πλήρες περιβάλλοντος ανάπτυξης, ώστε να παραχθεί ο κώδικας, ή ακόμα και τα διαγράμματα που θα υλοποιηθούν, μπορεί να χρησιμοποιηθεί η εγκατάσταση του Camunda μέσω Docker. Αυτή η διαδικασία επιτρέπει την εκτέλεση του Camunda ως ένα απομονωμένο περιβάλλον μέσω ενός Docker container, το οποίο μπορεί να είναι γρήγορο και ευέλικτο για την ανάπτυξη και τον έλεγχο του κώδικα ή των διαγραμμάτων που θα υλοποιηθούν σε εκείνο.

Η εγκατάσταση του Camunda, μέσω Docker, επιτρέπει την αμεσότητα του δημιουργημένου περιβάλλοντος μεταξύ των εργαζομένων μιας επιχείρησης. Επίσης, υπάρχει η δυνατότητα της διαχείρισης των εκδόσεων του Camunda με μεγαλύτερη ευκολία, αφού κάθε container περιέχει μια συγκεκριμένη έκδοση του. Επομένως, η δημιουργία γρήγορου και απομονωμένου περιβάλλοντος ανάπτυξης, προσφέρει αποτελεσματικότητα και υποστήριξη στους μηχανικούς πληροφορικής κατά τη φάση της ανάπτυξης των εφαρμογών που βασίζονται στο Camunda.

### Επεξήγηση της self-managed προσέγγισης στο Docker

Με τη self-managed προσέγγιση, υφίσταται πλήρης έλεγχος και ευελιξία στην εγκατάσταση και τη διαμόρφωση των Docker containers. Υπάρχει η δυνατότητα επιλογής των εικόνων Docker που θα χρησιμοποιηθούν, η διαμόρφωση των αντίστοιχων παραμέτρων για τη ρύθμιση του Camunda, όπως οι πόροι (μνήμη, CPU), οι πόρτες (ports), καθώς και οι αποθηκευτικοί χώροι (storage/volumes). Επιπλέον, με τη self-managed προσέγγιση, ο μηχανικός πληροφορικής είναι υπεύθυνος για την αναβάθμιση και τη διαχείριση των εκδόσεων των εικόνων Docker που θα χρησιμοποιήσει. Αυτό σημαίνει ότι, ο μηχανικός πληροφορικής μπορεί να ελέγχει ποιες εκδόσεις χρησιμοποιούνται στο περιβάλλον και να ενημερώνει σε νεότερες εκδόσεις όταν αυτό είναι απαραίτητο.

Η self-managed προσέγγιση απαιτεί εμπειρογνωμοσύνη και κατανόηση του Docker και των σχετικών ρυθμίσεων. Απαιτεί επίσης την αφιέρωση χρόνου και πόρων για τη διαχείριση των containers και την αντιμετώπιση προβλημάτων που μπορεί να προκύψουν. Παρόλα αυτά, η self-managed προσέγγιση προσφέρει τη μέγιστη ευελιξία και έλεγχο στην εγκατάσταση και τη

διαχείριση των Docker containers, επιτρέποντάς την διαμόρφωση του περιβάλλοντος σύμφωνα με τις ακριβείς απαιτήσεις της εφαρμογής.

- **Ελευθερία στην επιλογή του περιβάλλοντος:** Με την self-managed προσέγγιση, εγκαθίσταται το Camunda σε οποιοδήποτε on-premises περιβάλλον εξυπηρετεί καλύτερα τον μηχανικό πληροφορικής, ο οποίος διατηρεί πλήρη έλεγχο για την επιλογή του υλικού, του λειτουργικού συστήματος και τις ρυθμίσεις που ταιριάζουν καλύτερα στις απαιτήσεις του.
- **Προσαρμοσμένη αντιμετώπιση της ασφάλειας:** Με την self-managed προσέγγιση, ο μηχανικός πληροφορικής διαθέτει την πλήρη ευθύνη για την ασφάλεια των Docker container. Εφαρμόζει κατάλληλες πρακτικές ασφάλειας, προσαρμόζει τις ρυθμίσεις ασφαλείας και ελέγχει την πρόσβαση στο περιβάλλον σας.
- **Προσαρμογή και προετοιμασία για την παραγωγική χρήση:** Η self-managed προσέγγιση του μηχανικού πληροφορικής δίνει την ευκαιρία να προσαρμοστεί πλήρως το περιβάλλον για παραγωγική χρήση. Μπορεί κανείς να ρυθμίσει υψηλές διαθεσιμότητα, κλιμακωσιμότητα και αντοχή σε αυξημένο φορτίο, προετοιμάζοντας το περιβάλλον για παραγωγικές εφαρμογές.
- **Εκμετάλλευση επιπλέον λειτουργικότητας του Docker:** Με την self-managed προσέγγιση, υπάρχει η ευελιξία στην εκμετάλλευση όλων των δυνατοτήτων του Docker. Επίσης, υπάρχει η δυνατότητα δημιουργίας προσαρμοσμένων Docker images, η διαμόρφωση των ρυθμίσεων δικτύου, η διαχείριση των όγκων αποθήκευσης καθώς και άλλα.

## Οδηγίες εγκατάστασης Camunda 7 σε Docker

Για την εγκατάσταση του Camunda 7 σε ένα περιβάλλον Docker το οποίο θα λειτουργεί σε συστήματα "on premises" σύμφωνα με τις οδηγίες που προσφέρει το Camunda, θα πρέπει να ακολουθηθούν τα παρακάτω βήματα:

### Βήμα 1: Λήψη της Docker εικόνας του Camunda:

Για την έναρξη της διαδικασίας εγκατάστασης, πρέπει πρώτα να ληφθεί η επίσημη Docker εικόνα του Camunda από το Docker Hub. Για την λήψη της Docker εικόνας στον διακομιστή, χρησιμοποιείται η εντολή "docker pull". Η εικόνα περιλαμβάνει όλα τα απαραίτητα στοιχεία και ρυθμίσεις για την εκτέλεση της πλατφόρμας Camunda BPM μέσα σε ένα Docker container. Είναι σημαντικό να αναφερθεί ότι, ο μηχανικός πληροφορικής πρέπει να καθορίσει ποια έκδοση θα χρησιμοποιήσει. Η εικόνα Docker περιλαμβάνει τον διακομιστή εφαρμογής Camunda, τις web εφαρμογές και μια προ-ρυθμισμένη βάση δεδομένων, τα οποία διευκολύνουν τη διαδικασία εγκατάστασης.

### Βήμα 2: Δημιουργία ενός Docker Container:

Αφού ληφθεί η Docker εικόνα του Camunda, το επόμενο βήμα είναι να δημιουργηθεί ένα Docker container από αυτήν. Τα Docker containers είναι ελαφριά και απομονωμένα περιβάλλοντα που

ενθαρρύνουν την ενσωμάτωση της εφαρμογής του Camunda. Χρησιμοποιείται η εντολή "docker run" για αυτόν τον σκοπό. Αυτή η εντολή δημιουργεί έναν container για το Camunda με τις καθορισμένες παραμέτρους. Σε αυτήν την περίπτωση, η παράμετρος "-d" εκτελεί τον container σε αποσυνδεδεμένη λειτουργία, δηλαδή το «τρέχει» στο παρασκήνιο. Η παράμετρος "--name" αναθέτει ένα όνομα ("camunda" σε αυτό το παράδειγμα) στον container, καθιστώντας το πιο εύκολο στη διαχείριση. Τέλος, η παράμετρος "-p" αντιστοιχεί τη θύρα(port) 8080 από το container στον υπολογιστή μας. Η θύρα 8080 είναι η προεπιλεγμένη θύρα στην οποία λειτουργεί το Camunda για τις web εφαρμογές του. Η αντιστοίχιση των θυρών μπορεί να προσαρμοστεί ανάλογα με τις απαιτήσεις του μηχανικού.

### **Βήμα 3: Πρόσβαση στις web εφαρμογές του Camunda:**

Εφόσον τεθεί σε λειτουργία το Docker container του Camunda, αποκτάται πρόσβαση στις web εφαρμογές του Camunda μέσω ενός web browser, εισάγοντας τη διεύθυνση URL <http://localhost:8080/camunda/app/welcome/default/> στη γραμμή διευθύνσεων του προγράμματος περιήγησής. Αυτή η διεύθυνση URL οδηγεί στην αρχική σελίδα του Camunda, όπου δύναται κανείς πλέον να συνδεθεί και να χρησιμοποιήσει την συγκεκριμένη πλατφόρμα BPM.

Τα προεπιλεγμένα διαπιστευτήρια για την είσοδο μας στην εφαρμογή σύμφωνα με τις αρχικές οδηγίες εγκατάστασης του Camunda ορίζεται ως όνομα χρήστη "demo" και τον κωδικό πρόσβασης ως "demo". Ωστόσο, για λόγους ασφαλείας, είναι συνετό να αλλαχθούν τα προεπιλεγμένα διαπιστευτήρια.

### **Βήμα 4: Ανάπτυξη Διαδικασιών:**

Μετά την απόκτηση πρόσβασης στις web εφαρμογές του Camunda, ο μηχανικός πληροφορικής αναπτύσσει μοντέλα διαδικασίας BPMN (Business Process Model and Notation). Αφού τα μοντέλα των διαδικασιών είναι έτοιμα, «ανεβαίνουν» και αναπτύσσονται μέσω των web εφαρμογών του Camunda. Ο Camunda παρέχει χρήσιμες δι-επαφές χρήστη για τη διαχείριση και ανάπτυξη των διαδικασιών. Αυτό το βήμα επιτρέπει την εισαγωγή των επιχειρηματικών διαδικασιών εντός της πλατφόρμας Camunda, αυτοματοποιώντας τις ροές εργασίας.

Το Camunda Modeler ή άλλα συμβατικά εργαλεία μοντελοποίησης BPMN μπορούν να χρησιμοποιηθούν για το σχεδιασμό και τη δημιουργία των διαγραμμάτων των διαδικασιών.

### **Βήμα 5: Παρακολούθηση και Διαχείριση των Διαδικασιών:**

Η εφαρμογή Cockpit του Camunda έχει σχεδιαστεί για την πραγματική παρακολούθηση και διαχείριση των εκτελούμενων διαδικασιών. Το Cockpit επιτρέπει την παρακολούθηση της προόδου των διαδικασιών, την αναγνώριση των φραγμών και την παρέμβαση, όπως ανακατανομή εργασιών ή προσαρμογή μεταβλητών των διαδικασιών.

Επιπλέον, το Cockpit χρησιμοποιείται για να βελτιωθούν οι διαδικασίες, λαμβάνοντας αποφάσεις με βάση τα δεδομένα που παρουσιάζονται στον πίνακα ελέγχου. Η δυνατότητα παρακολούθησης

και διαχείρισης των διαδικασιών σε πραγματικό χρόνο είναι ένα σημαντικό χαρακτηριστικό του Camunda, που συμβάλλει στην αποτελεσματική αυτοματοποίηση των διαδικασιών.

## Βήμα 6: Ασφάλεια την Εγκατάστασή σας (Προαιρετικά):

Εάν και είναι προαιρετικό το βήμα, είναι εξίσου σημαντικό να ληφθεί υπόψη η ασφάλεια της εγκατάστασης, ειδικά αν πρόκειται να χρησιμοποιηθεί το Docker σε περιβάλλον παραγωγής.

Για την ενίσχυση της ασφάλειας της εγκατάστασής του Camunda, μπορούν να ληφθούν διάφορα μέτρα. Για παράδειγμα, κατά τη διάρκεια της παραμετροποίησης της εφαρμογής, είναι καλό να χρησιμοποιηθούν μόνο ασφαλείς συνδέσεις HTTPS, και όλες τις υπόλοιπες που δεν εμπίπτουν σε αυτό το πρωτόκολλο, να τις απορρίπτει.

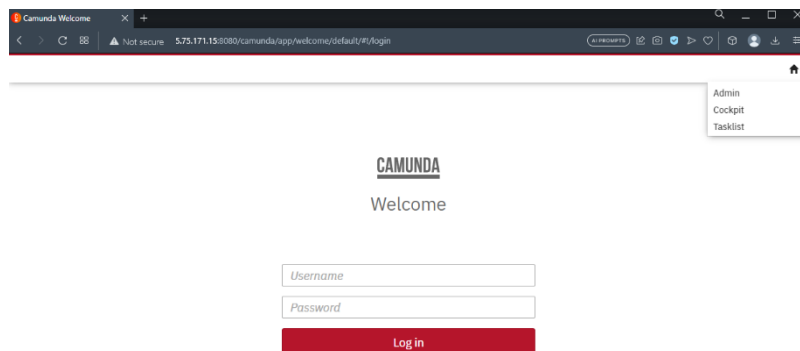
Η διασφάλιση ότι η εγκατάσταση του Camunda τηρεί όλα τα πρωτόκολλα ασφαλείας είναι ζωτικής σημασίας για την προστασία των επιχειρηματικών διαδικασιών, καθώς και των ευαίσθητων πληροφοριών από μη εξουσιοδοτημένη πρόσβαση ή παραβιάσεις δεδομένων. Το Camunda στον οδηγό εγκατάσταση παρέχει τεκμηριωμένες και αναλυτικές οδηγίες για την ασφάλεια της εγκατάστασής, οι οποίες πρέπει να ακολουθηθούν πιστά.

```
root@camunda7:~# docker ps -a
CONTAINER ID   IMAGE                                COMMAND                  CREATED        STATUS        PORTS                                                                                               NAMES
ca4b2f5a1a07   camunda/camunda-bpm-platform:latest  "/sbin/tini -- ./cam..."  5 weeks ago   Up 5 weeks   0.0.0.0:8080->8080/tcp, :::8080->8080/tcp, 8080/tcp, 9404/tcp, 0.0.0.0:8443->8443/tcp, :::8443->8443/tcp  camunda_camunda_1
root@camunda7:~#
```

Εικόνα 17 Απεικόνιση Επιτυχούς Δημιουργίας Docker δοχείου

## Βήμα 7:

Ανοίγοντας έναν περιηγητή και μεταβαίνοντας στη διεύθυνση <http://localhost:8080/camunda>, εμφανίζεται το περιβάλλον του Camunda.



Εικόνα 18 Απεικόνιση Οθόνης Σύνδεσης στο Camunda

Τα παραπάνω βήματα θα εγκαταστήσουν το Camunda 7 σε ένα Docker container και θα το κάνουν προσβάσιμο μέσω του περιηγητή. Ωστόσο, πρέπει να επιβεβαιωθεί ότι δεν υπάρχει

κάποιος άλλος διακομιστής που εκτελείται στη θύρα 8080 του συστήματος. Εάν η θύρα 8080 είναι ήδη σε χρήση, μπορεί να αλλαχθεί η θύρα στην εντολή του Docker ως εξής: `-p <νέα_θύρα>:8080`, όπου `<νέα_θύρα>` είναι προς χρήση.

## Υλοποίηση εφαρμογών

Για την υλοποίηση των εφαρμογών που θα παρέχουν πρόσβαση στους χρήστες των δημόσιων φορέων και θα τους επιτρέπουν να διαχειρίζονται τις διαδικασίες και τα αιτήματά τους, αναπτύχθηκαν δύο διαφορετικές εφαρμογές.

Η πρώτη από αυτές αφορά το περιβάλλον όπου οι χρήστες εισέρχονται και αλληλοεπιδρούν με το σύστημα εφεξής FrontEnd, ενώ η δεύτερη, εφεξής BackEnd, λειτουργεί ως ενδιάμεσος παράγοντας μεταξύ του FrontEnd και της πλατφόρμας του Camunda.

Η πρώτη εφαρμογή έχει σχεδιαστεί με στόχο να προσφέρει μια φιλική και ευανάγνωστη διεπαφή για τους χρήστες. Αυτό το περιβάλλον επιτρέπει στους χρήστες να συνδεθούν, να εξερευνήσουν τις διαθέσιμες λειτουργίες και να υποβάλουν τα αιτήματά τους. Επιπλέον, παρέχει τη δυνατότητα παρακολούθησης της προόδου των αιτημάτων τους και της εκτέλεσης των διαδικασιών.

Η δεύτερη λειτουργεί ως μεσολαβητής ανάμεσα στο FrontEnd και την πλατφόρμα του Camunda. Αναλαμβάνει τον ρόλο του διαμεσολαβητή που διαχειρίζεται τα αιτήματα των χρηστών, επικοινωνεί με το Camunda για την εκτέλεση των διαδικασιών και επιστρέφει τα αποτελέσματα στο FrontEnd. Αυτή η εφαρμογή διασφαλίζει τη σωστή λειτουργία και τον συντονισμό μεταξύ του χρήστη και της πλατφόρμας Camunda, προσφέροντας ένα ομαλό και αποτελεσματικό ροή εργασίας.

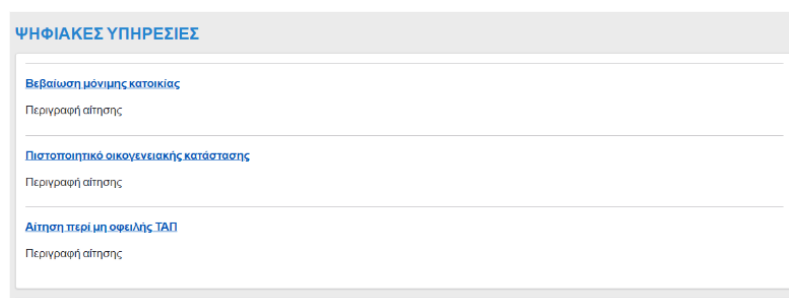
Με αυτόν τον τρόπο, οι δημόσιοι φορείς μπορούν να προσφέρουν μια προηγμένη εμπειρία στους χρήστες τους, επιτρέποντάς τους να διεκπεραιώνουν τα αιτήματά τους με αποτελεσματικότητα και αξιοπιστία, ενώ ταυτόχρονα διασφαλίζουν τη σωστή εκτέλεση των διαδικασιών μέσω της πλατφόρμας Camunda.

### FrontEnd

Για την υλοποίηση της FrontEnd εφαρμογής χρησιμοποιήθηκε η ReactJS, η οποία αποτελεί μια προηγμένη βιβλιοθήκη JavaScript που αξιοποιείται με σκοπό τη δημιουργία υψηλής ποιότητας διεπαφών χρήστη για ιστοσελίδες και εφαρμογές. Η βιβλιοθήκη αυτή βασίζεται σε λειτουργικά στοιχεία που συνδυάζονται για να δημιουργήσουν την εφαρμογή και είναι ανοικτού κώδικα, προσφέροντας τη δυνατότητα ελεύθερης χρήσης. Κεντρίζει την προσοχή του κυρίως στην ανάπτυξη διαδικτυακών και κινητών εφαρμογών, επιτρέποντας την αυτόματη ενημέρωση της διεπαφής χρήστη όταν αλλάζει η κατάσταση της εφαρμογής, χωρίς την ανάγκη για ανανέωση της σελίδας. Είναι επίσης η προτιμώμενη επιλογή των προγραμματιστών για την ανάπτυξη μεγάλων και πολύπλοκων εφαρμογών, λόγω της απλότητας χρήσης που προσφέρει.

Κατά τη διάρκεια της ανάπτυξης, επιλέχθηκε να χρησιμοποιηθεί η Typescript (TSX). Η ενσωμάτωση της TSX στο React προσφέρει σημαντικά πλεονεκτήματα στους προγραμματιστές, όπως την ενίσχυση του ελέγχου τύπων, το οποίο βοηθά στην αποφυγή τυπικών σφαλμάτων κατά την εκτέλεση του κώδικα, ενώ ταυτόχρονα βελτιώνει τη συντηρησιμότητα του κώδικα.

Η αρχική σελίδα αυτής της πλατφόρμας διακρίνεται σε δύο κύρια τμήματα, γνωστά και ως στοιχεία (components). Το πρώτο από αυτά παρουσιάζει τις διαθέσιμες αιτήσεις που μπορούν να επιλέξουν οι χρήστες και να προχωρήσουν σε μια διαδικασία. Σημειώνεται ότι όλα τα αιτήματα που αναφέρονται σε αυτό το τμήμα περιγράφονται ως αιτήματα στο Camunda, αν και, όπως αναφέρθηκε προηγουμένως, πραγματοποιούνται όλα μέσω του Back-End Web API. Όταν ο χρήστης επιλέγει μια από τις διαθέσιμες αιτήσεις, ξεκινά μια διαδικασία εκτελώντας ένα αίτημα προς το Camunda, το οποίο εκπροσωπεί την αντίστοιχη διαδικασία.



Εικόνα 19 Απεικόνιση Ψηφιακών Υπηρεσιών

Το δεύτερο στοιχείο (component), γνωστό ως "Οι Αιτήσεις μου" (MyApplications), λειτουργεί ως μια συνοπτική προβολή των αιτήσεων που σχετίζονται με ένα συγκεκριμένο χρήστη. Αυτή η προβολή παρουσιάζει τις αιτήσεις που έχουν ήδη ολοκληρωθεί επιτυχώς, αυτές που έχουν απορριφθεί, καθώς και αυτές που είναι ενεργές και προσωρινά αποθηκευμένες. Για να επιτευχθεί αυτή η προβολή, απαιτείται μια ακολουθία από αιτήματα προς το σύστημα Camunda.

Αρχικά, δημιουργούνται δύο καρτέλες (tabs) που περιέχουν τις ολοκληρωμένες και τις ενεργές αιτήσεις του χρήστη αντίστοιχα. Αυτό πραγματοποιείται μέσω αιτημάτων προς το Camunda. Για τις ολοκληρωμένες αιτήσεις, πραγματοποιείται αρχικά ένα αίτημα που επιστρέφει τις διαδικασίες που έχουν ολοκληρωθεί.

Στη συνέχεια, για την παρουσίαση των αποτελεσμάτων που απαιτούνται, εφαρμόζονται φίλτρα στα δεδομένα. Αυτή η διαδικασία περιλαμβάνει την επιστροφή των διαδικασιών (στη συγκεκριμένη περίπτωση, των αιτήσεων) που αντιστοιχούν στο συγκεκριμένο όνομα χρήστη και έχουν ολοκληρωθεί, με την ταξινόμησή τους από την πιο πρόσφατη προς την παλαιότερη. Στη συνέχεια, γίνεται ένα επιπλέον αίτημα προς το Camunda για να λάβουν τα απαιτούμενα δεδομένα για κάθε μία από τις αιτήσεις που ανακύπτουν από το προηγούμενο βήμα. Αντίστοιχες διαδικασίες εφαρμόζονται και για τις ενεργές αιτήσεις του χρήστη, με τις κατάλληλες τροποποιήσεις στα αιτήματα προκειμένου να αντικατοπτρίζουν τα δεδομένα που απαιτούνται για τις ενεργές αιτήσεις.

Ολοκληρωμένες αιτήσεις		Ενεργές αιτήσεις (citizen)	
Όνομα Αίτησης	Κατάσταση	Κωδικός	
Πιστοποιητικό οικογενειακής κατάστασης	Ολοκληρωμένη	c3d67e07-f7a0-437d-88c6-59f9a296edfa	📄

Εικόνα 20 Απεικόνιση Λίστας Ολοκληρωμένων και Ενεργών Αιτήσεων

## BackEnd

Για την υλοποίηση του backend Api χρησιμοποιήθηκε η τεχνολογία *.net core web api*. Το .NET Core είναι ένα ανοιχτού κώδικα πλαίσιο ανάπτυξης λογισμικού από την Microsoft, το οποίο παρέχει μια πλατφόρμα για τη δημιουργία διαδικτυακών εφαρμογών και web services [1]. Το .NET Core WebAPI είναι μια συγκεκριμένη επέκταση του .NET Core, η οποία επιτρέπει την ανάπτυξη διαδικτυακών API. Το .NET Core Web API παρέχει πολλά πλεονεκτήματα στους προγραμματιστές. Μερικά από αυτά είναι:

- **Ανεξαρτησία πλατφόρμας:** Το .NET Core είναι διαθέσιμο για πολλές πλατφόρμες, συμπεριλαμβανομένων των Windows, Linux και macOS [2]. Αυτό σημαίνει ότι οι προγραμματιστές μπορούν να αναπτύξουν εφαρμογές σε οποιαδήποτε πλατφόρμα επιθυμούν.
- **Υψηλή απόδοση:** Το .NET Core Web API είναι σχεδιασμένο για να παρέχει υψηλή απόδοση στις διαδικτυακές εφαρμογές και στα web services [3].
- **Ασφάλεια:** Το .NET Core Web API παρέχει εξαιρετικές δυνατότητες ασφάλειας, όπως αυθεντικοποίηση και εξουσιοδότηση [4].
- **Ευκολία στην ανάπτυξη:** Το .NET Core WebAPI είναι εύκολο στη χρήση και στην ανάπτυξη, με ένα ευέλικτο σύστημα δρομολόγησης, το οποίο επιτρέπει στους προγραμματιστές να δημιουργούν και να προσαρμόζουν εύκολα τα διαδραστικά στοιχεία των εφαρμογών [5].

Επιπλέον, το .NET Core WebAPI υποστηρίζει ανοιχτά πρότυπα και πρωτόκολλα όπως τα JSON και RESTful APIs, και μπορεί να χρησιμοποιηθεί με μια ποικιλία εργαλείων και βιβλιοθηκών ανάπτυξης. Συνολικά, το .NET Core WebAPI είναι ένα ισχυρό εργαλείο για τη δημιουργία διαδικτυακών εφαρμογών και web services. Η ανεξαρτησία πλατφόρμας, η υψηλή απόδοση, η ασφάλεια και η ευκολία στην ανάπτυξη καθιστούν το .NET Core Web API μια εξαιρετική επιλογή για τους προγραμματιστές που αναζητούν ένα αποδοτικό και αξιόπιστο πλαίσιο ανάπτυξης διαδικτυακών εφαρμογών.

Συνολικά, ο παρακάτω κώδικας παρέχει ένα API για την επικοινωνία της Frontend εφαρμογής με το Camunda. Αυτό το API εκτελεί αιτήσεις προς το Camunda REST API και επιστρέφει τα αποτελέσματα στον χρήστη μέσω των μεθόδων που περιγράφηκαν προηγουμένως. Με αυτόν τον τρόπο, ο χρήστης μπορεί να αλληλεπιδράσει με το Camunda και να λαμβάνει δεδομένα σχετικά με διεργασίες, εργασίες, μεταβλητές και πληροφορίες χρήστη.

Ως τούτου δημιουργήθηκε ένα Service με το όνομα "CamundaService" που παρέχει διάφορες μεθόδους για την αλληλεπίδραση με την πλατφόρμα Camunda BPM χρησιμοποιώντας το Camunda REST API. Ας αναλύσουμε τον κώδικα βήμα προς βήμα:

### 1. Κατασκευαστής (Constructor):

- Γίνεται αρχικοποίηση ενός αντικείμενου HttpClient και το ρυθμίζει με βάση τα παρεχόμενα δεδομένα (camundaApiUrl, camundaUsername, camundaPassword).

Ορίζει τη βάση της διεύθυνσης URL και τον τίτλο εξουσιοδότησης στην κεφαλίδα των αιτημάτων ως βασική εξουσιοδότηση (Basic Authorization).

2. Μέθοδος "StartProcessInstance":
  - Δέχεται ένα κλειδί (processDefinitionKey) ως παράμετρο.
  - Αποστέλλει μια POST αίτηση στην κατάλληλη διεύθυνση API για να ξεκινήσει μια νέα διεργασία με βάση το κλειδί του ορισμού διεργασίας.
  - Επιστρέφει το αποτέλεσμα της αίτησης ως string.
3. Μέθοδοι "GetList", "GetList2", "GetList3":
  - Δέχονται διάφορες παραμέτρους (businessKey, involvedUser, candidateGroup) αντίστοιχα.
  - Αποστέλλουν GET αιτήματα στο API για να ανακτήσουν λίστες εργασιών με βάση τις παρεχόμενες παραμέτρους.
  - Επιστρέφουν τις λίστες εργασιών ως λίστες αντικειμένων τύπου "Class1".
4. Μέθοδοι "GetProcessDefinitions", "GetProcessDefinitionsByGroup", "GetActiveProcessDefinitionsByGroup", "GetSuspendedProcessDefinitionsByGroup":
  - Δέχονται παραμέτρους (group) αντίστοιχα.
  - Αποστέλλουν GET αιτήματα στο API για να ανακτήσουν ορισμούς διεργασιών με βάση τις παρεχόμενες παραμέτρους.
  - Επιστρέφουν τους ορισμούς διεργασιών ως λίστες αντικειμένων τύπου "ProcessDefinition".
5. Μέθοδος "GetFormKey":
  - Δέχεται ένα αναγνωριστικό εργασίας (taskId) ως παράμετρο.
  - Αποστέλλει GET αίτημα στο API για να ανακτήσει το κλειδί φόρμας για το συγκεκριμένο taskId.
  - Επιστρέφει το αποτέλεσμα ως αντικείμενο τύπου "FormKey".
6. Μέθοδοι "GetUserCompletedInstances", "GetUserActiveInstances":
  - Δέχονται ένα αναγνωριστικό χρήστη (userId) ως παράμετρο.
  - Αποστέλλουν GET αιτήματα στο API για να ανακτήσουν ολοκληρωμένες ή ενεργές περιπτώσεις διεργασίας για τον συγκεκριμένο χρήστη.
  - Επιστρέφουν τις περιπτώσεις ως λίστες αντικειμένων τύπου "CompletedInstance" ή "ActiveInstance" αντίστοιχα.
7. Μέθοδοι "GetTaskVariables", "GetCompletedTaskVariables", "GetActiveTaskVariables":
  - Δέχονται παραμέτρους (taskId, processInstanceId) αντίστοιχα.
  - Αποστέλλουν GET αιτήματα στο API του Camunda για να ανακτήσουν μεταβλητές εργασιών ή μεταβλητές που σχετίζονται με ολοκληρωμένες ή ενεργές περιπτώσεις διεργασίας.
  - Επιστρέφουν τις μεταβλητές ως λεξικά αντικειμένων τύπου "VariableInfo", "VarClass" ή "Dictionary<string, object>" αντίστοιχα.
8. Μέθοδος "GetUserInfo":



- Δέχεται ένα αναγνωριστικό χρήστη (user) ως παράμετρο.
- Αποστέλλει GET αίτημα στο API του Camunda για να ανακτήσει πληροφορίες χρήστη για τον συγκεκριμένο χρήστη.
- Επιστρέφει τις πληροφορίες ως αντικείμενο τύπου "User".

#### 9. Μέθοδος "GetTaskById":

- Δέχεται ένα αναγνωριστικό εργασίας (taskId) ως παράμετρο.
- Αποστέλλει GET αίτημα στο API του Camunda για να ανακτήσει τις πληροφορίες μιας εργασίας με βάση το taskId.
- Επιστρέφει τις πληροφορίες ως αντικείμενο τύπου "Class1".

#### 10. Μέθοδοι "SuspendProcess", "ActivateProcess":

- Δέχονται ένα αναγνωριστικό διεργασίας (processId) ως παράμετρο.
- Αποστέλλουν PUT αιτήματα στο API για να αναστείλουν ή να ενεργοποιήσουν μια διεργασία με βάση το processId.

#### 11. Μέθοδοι "ClaimTask", "UnclaimTask":

- Δέχονται παραμέτρους (taskId, userId) αντίστοιχα.
- Αποστέλλουν POST αιτήματα στο API του Camunda για να αξιοποιήσουν ή να απελευθερώσουν μια εργασία με βάση το taskId και τον χρήστη με τον αντίστοιχο userId.

#### 12. Μέθοδος "CompleteTask":

- Δέχεται παραμέτρους (taskId, completeRequest) όπου completeRequest είναι ένα αντικείμενο τύπου "CompleteRequest".
- Αποστέλλει POST αίτημα στο API του Camunda για να ολοκληρώσει μια εργασία με βάση το taskId και το completeRequest.
- Επιστρέφει το αποτέλεσμα της αίτησης ως λεξικό αντικειμένων τύπου "Dictionary<string, object>".

#### 13. Μέθοδος "StartProcess":

- Δέχεται παραμέτρους (processDefinitionId, startRequest) όπου startRequest είναι ένα αντικείμενο τύπου "StartRequest".
- Αποστέλλει POST αίτημα στο API για να ξεκινήσει μια διεργασία με βάση το processDefinitionId και το startRequest.
- Επιστρέφει το αποτέλεσμα της αίτησης ως αντικείμενο τύπου "StartResponse".

Συνολικά, η κλάση CamundaService παρέχει μια σειρά από μεθόδους που επικοινωνούν με το Camunda BPM API και επιτρέπουν την εκτέλεση διάφορων εργασιών όπως ξεκίνημα διεργασιών, ανάκτηση λιστών εργασιών, ανάκτηση μεταβλητών, αλλαγή καταστάσεων διεργασιών, ανάκτηση πληροφοριών χρηστών και ολοκλήρωση εργασιών. Ο κώδικας επίσης χρησιμοποιεί τη βιβλιοθήκη Json.NET (Newtonsoft.Json) για την ανάλυση και αποτύπωση αντικειμένων JSON.

Συνολικά, ο κώδικας παρουσιάζει μια υλοποίηση ενός wrapper για το Camunda BPM API που προσφέρει ποικίλες λειτουργίες για τη διαχείριση διεργασιών, εργασιών και μεταβλητών σε ένα σύστημα BPM.

Στην συνέχεια δημιουργηθήκαν κλάσης όπου ορίζονται συγκεκριμένα μοντέλα δεδομένων που χρησιμοποιούνται στο πλαίσιο της ανάπτυξης του api. Αυτά τα μοντέλα δεδομένων χρησιμοποιούνται για την ανταλλαγή πληροφοριών μεταξύ της εφαρμογής και του Camunda BPM. Ας αναλύσουμε τον κώδικα βήμα προς βήμα:

1. Κλάση "AuthenticationRequest":
  - Περιέχει δύο ιδιότητες: "Firstname" και "Password".
  - Χρησιμοποιείται για τον έλεγχο ταυτότητας με την υπηρεσία πιστοποίησης.
2. Κλάση "CamundaModel":
  - Περιέχει μια σειρά από εσωτερικές κλάσεις που περιγράφουν τα μοντέλα δεδομένων που χρησιμοποιούνται στην εφαρμογή.
  - Κάθε εσωτερική κλάση αντιστοιχεί σε ένα συγκεκριμένο μοντέλο δεδομένων.
3. Κλάση "CompleteTask":
  - Περιέχει τις ιδιότητες "variables" και "withVariablesInReturn".
  - Η ιδιότητα "variables" είναι ένα λεξικό (Dictionary) που περιέχει μια σειρά από μεταβλητές.
  - Η ιδιότητα "withVariablesInReturn" είναι ένα boolean που υποδηλώνει εάν οι μεταβλητές θα επιστραφούν μετά την ολοκλήρωση της εργασίας.
4. Κλάση "CompleteRequest":
  - Περιέχει τις ιδιότητες "variables" και "withVariablesInReturn".
  - Η ιδιότητα "variables" είναι ένα λεξικό (Dictionary) που περιέχει μια σειρά από μεταβλητές.
  - Η ιδιότητα "withVariablesInReturn" είναι ένα boolean που υποδηλώνει εάν οι μεταβλητές θα επιστραφούν μετά την ολοκλήρωση της εργασίας.
5. Κλάση "Variable":
  - Περιέχει την ιδιότητα "value".
  - Χρησιμοποιείται για την αναπαράσταση μιας μεταβλητής με την τιμή της.
6. Κλάση "ResponseModel":
  - Περιέχει την ιδιότητα "variables" που είναι ένα λεξικό (Dictionary) που περιέχει μια σειρά από μεταβλητές.
7. Κλάση "StartRequest":
  - Περιέχει τις ιδιότητες "variables" και "businessKey".
  - Η ιδιότητα "variables" είναι ένα λεξικό (Dictionary) που περιέχει μια σειρά από μεταβλητές.
  - Η ιδιότητα "businessKey" περιέχει τον επιχειρηματικό κλειδί για την έναρξη της διεργασίας.
8. Κλάση "StartResponse":

- Περιέχει τις ιδιότητες "links", "id", "definitionId", "businessKey", "caseInstanceId", "ended", "suspended" και "tenantId".
- Παρέχει πληροφορίες για την έναρξη μιας διεργασίας.

#### 9. Κλάση "Link":

- Περιέχει τις ιδιότητες "method", "href" και "rel".
- Χρησιμοποιείται για την αναπαράσταση ενός συνδέσμου.

#### 10. Κλάση "User":

- Περιέχει τις ιδιότητες "id", "firstName", "lastName", "fullName", "emailAddress", "password" και "group".
- Χρησιμοποιείται για την αναπαράσταση ενός χρήστη του συστήματος. Περιλαμβάνει πληροφορίες όπως το ID του χρήστη, το όνομα, το επίθετο, το πλήρες όνομα, το email, τον κωδικό πρόσβασης και την ομάδα στην οποία ανήκει.

#### 11. Κλάση "VariableInfo":

- Περιέχει τις ιδιότητες "type", "value" και "valueInfo".
- Χρησιμοποιείται για την αναπαράσταση πληροφοριών μιας μεταβλητής.

#### 12. Κλάση "RootObject":

- Περιέχει την ιδιότητα "property1" που είναι ένας πίνακας από αντικείμενα τύπου "VarClass".
- Χρησιμοποιείται για την αναπαράσταση μιας ρίζας αντικειμένου με μεταβλητές.

#### 13. Κλάση "VarClass":

- Περιέχει τις ιδιότητες "type", "value", "valueInfo", "id", "name", "processDefinitionKey", "processDefinitionId", "processInstanceId", "executionId", "activityInstanceId", "caseDefinitionKey", "caseDefinitionId", "caseInstanceId", "caseExecutionId", "taskId", "errorMessage", "tenantId", "state", "createTime", "removalTime" και "rootProcessInstanceId".
- Περιγράφει μια μεταβλητή και τις σχετικές της πληροφορίες.

Στην συνέχεια της ανάπτυξης του API δημιουργήθηκε ένας Controller ο οποίος είναι μέρος του μοντέλου της αρχιτεκτονικής MVC (Model-View-Controller). Ο ρόλος του "Controller" είναι να αντιμετωπίζει τις αιτήσεις (requests) που λαμβάνονται από τους πελάτες (clients) και να αντιδρά σε αυτές, εκτελώντας τις κατάλληλες ενέργειες. Συνήθως, ο "Controller" λαμβάνει τις αιτήσεις από τον δρομολογητή (Router), και στη συνέχεια εκτελεί την κατάλληλη μέθοδο για να αντιμετωπίσει την αίτηση.

Μέσω του Controller, επιτυγχάνεται η αλληλεπίδραση με το μοντέλο και την υπηρεσία της εφαρμογής, ανακτώντας ή ενημερώνοντας δεδομένα, εκτελώντας λειτουργίες και αποστέλλοντας απαντήσεις πίσω στον πελάτη.

Παρακάτω, αναλύεται ο κώδικας βήμα προς βήμα:

#### Ελεγκτής "CamundaController":

Ο παραπάνω κώδικας ανήκει σε έναν ASP.NET Core Web API Controller με το όνομα **CamundaController**. όπου διαθέτει μια σειρά από δρομολογημένα (routed) endpoints που επιτρέπουν την επικοινωνία με το Camunda BPMN Engine μέσω της κλάσης **CamundaService**. Αναλύοντας τον κώδικα πιο διεξοδικά:

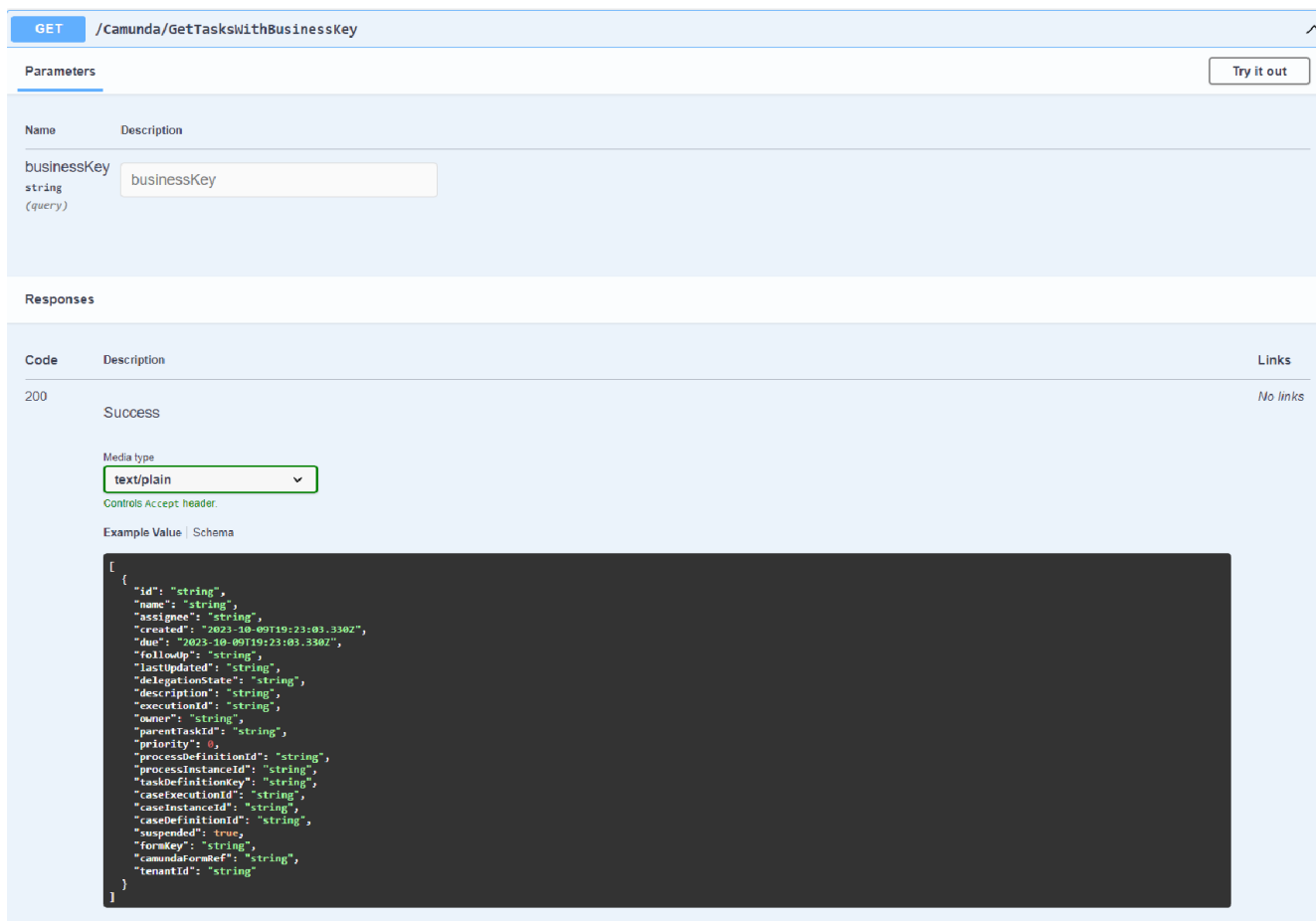
**Constructor:** Ο μηχανισμός που αναλαμβάνει να εγκαταστήσει την υπηρεσία **CamundaService** χρησιμοποιώντας «dependency injection». Η υπηρεσία αυτή χρησιμοποιείται για να εκτελεί λειτουργίες στο Camunda Engine.

### HTTP GET Endpoints:

Κάθε μέθοδος HTTP GET αντιστοιχεί σε ένα endpoint που επιτρέπει την ανάκτηση πληροφοριών από το Camunda Engine. Οι μέθοδοι περιλαμβάνουν τα εξής:

#### ➤ **GetTasksWithBusinessKey**

Όπως φαίνεται στην εικόνα, λαμβάνονται εργασίες βάσει ενός business key.



The screenshot displays a REST client interface for the endpoint `GET /Camunda/GetTasksWithBusinessKey`. Under the **Parameters** tab, a query parameter `businessKey` (string) is defined with a text input field containing the value `businessKey`. A `Try it out` button is visible. The **Responses** section shows a `200` `Success` response with a `text/plain` media type. Below the response, there are tabs for `Example Value` and `Schema`. The `Example Value` tab is active, showing a JSON object with the following structure:

```
[
  {
    "id": "string",
    "name": "string",
    "assignee": "string",
    "created": "2023-10-09T19:23:03.330Z",
    "due": "2023-10-09T19:23:03.330Z",
    "followup": "string",
    "lastUpdated": "string",
    "delegationState": "string",
    "description": "string",
    "executionId": "string",
    "owner": "string",
    "parentTaskId": "string",
    "priority": 0,
    "processDefinitionId": "string",
    "processInstanceId": "string",
    "taskDefinitionKey": "string",
    "caseExecutionId": "string",
    "caseInstanceId": "string",
    "caseDefinitionId": "string",
    "suspended": true,
    "formKey": "string",
    "camundaFormRef": "string",
    "tenantId": "string"
  }
]
```

Εικόνα 21 Απεικόνιση Οθόνης Κώδικα με "Business Key"

#### ➤ **GetTasksWithUser**

Λαμβάνονται εργασίες στις οποίες εμπλέκεται ένας συγκεκριμένος χρήστης.

GET /Camunda/GetTasksWithUser

Parameters Try it out

Name	Description
involvedUser string (query)	<input type="text" value="involvedUser"/>

Responses

Code	Description	Links
200	Success	No links

Media type:

Controls: Accept header:

Example Value | Schema

```

{
  "id": "string",
  "name": "string",
  "assignee": "string",
  "created": "2023-10-09T19:30:31.078Z",
  "due": "2023-10-09T19:30:31.078Z",
  "followUp": "string",
  "lastUpdated": "string",
  "delegationState": "string",
  "description": "string",
  "executionId": "string",
  "owner": "string",
  "parentTaskId": "string",
  "priority": 0,
  "processDefinitionId": "string",
  "processInstanceId": "string",
  "taskDefinitionKey": "string",
  "caseExecutionId": "string",
  "caseInstanceId": "string",
  "caseDefinitionId": "string",
  "suspended": true,
  "formKey": "string",
  "camundaFormRef": "string",
  "tenantId": "string"
}

```

Εικόνα 22 Απεικόνιση Οθόνης Κώδικα με "Συγκεκριμένο Χρήστη"

## ➤ GetTasksWithGroup

Λαμβάνονται εργασίες που ανήκουν σε μια ομάδα υποψηφίων.

Parameters

Name	Description
candidateGroup string (query)	candidateGroup

Responses

Code	Description	Links
200	Success	No links

Media type: text/plain

Example Value | Schema

```
{
  "id": "string",
  "name": "string",
  "assignee": "string",
  "created": "2023-10-09T19:30:47.486Z",
  "due": "2023-10-09T19:30:47.486Z",
  "followup": "string",
  "lastUpdated": "string",
  "delegationState": "string",
  "description": "string",
  "executionId": "string",
  "owner": "string",
  "parentTaskId": "string",
  "priority": 0,
  "processDefinitionId": "string",
  "processInstanceId": "string",
  "taskDefinitionKey": "string",
  "taskDefinitionId": "string",
  "caseInstanceId": "string",
  "caseDefinitionId": "string",
  "suspended": true,
  "formKey": "string",
  "camundaFormRef": "string",
  "tenantId": "string"
}
```

Εικόνα 23 Απεικόνιση Οθόνης Κώδικα με "Ομάδα Υποψηφίων"

## ➤ GetProcessDefinitions

Λαμβάνει τον ορισμό των διαδικασιών.

GET /Camunda/GetProcessDefinitions

Parameters

No parameters

Responses

Code	Description	Links
200	SUCCESS	No links

Media type: text/plain

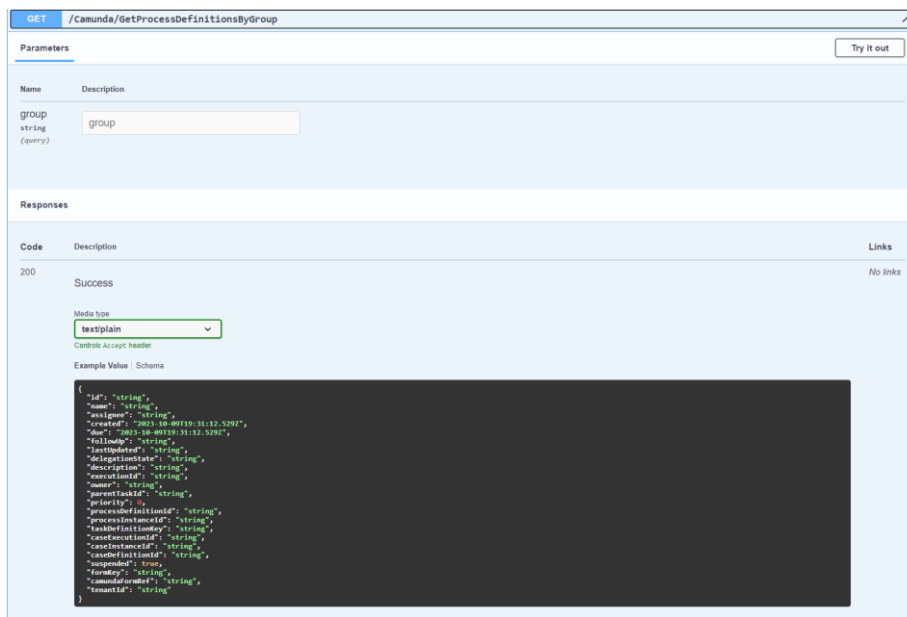
Example Value | Schema

```
{
  "id": "string",
  "name": "string",
  "assignee": "string",
  "created": "2023-10-09T19:31:01.358Z",
  "due": "2023-10-09T19:31:01.358Z",
  "followup": "string",
  "lastUpdated": "string",
  "delegationState": "string",
  "description": "string",
  "executionId": "string",
  "owner": "string",
  "parentTaskId": "string",
  "priority": 0,
  "processDefinitionId": "string",
  "processInstanceId": "string",
  "taskDefinitionKey": "string",
  "taskDefinitionId": "string",
  "caseInstanceId": "string",
  "caseDefinitionId": "string",
  "suspended": true,
  "formKey": "string",
  "camundaFormRef": "string",
  "tenantId": "string"
}
```

Εικόνα 24 Απεικόνιση Οθόνης Κώδικα με "Ορισμό Διαδικασιών"

## GetProcessDefinitionsByGroup

Λαμβάνει τον ορισμό των διαδικασιών για μια συγκεκριμένη ομάδα.



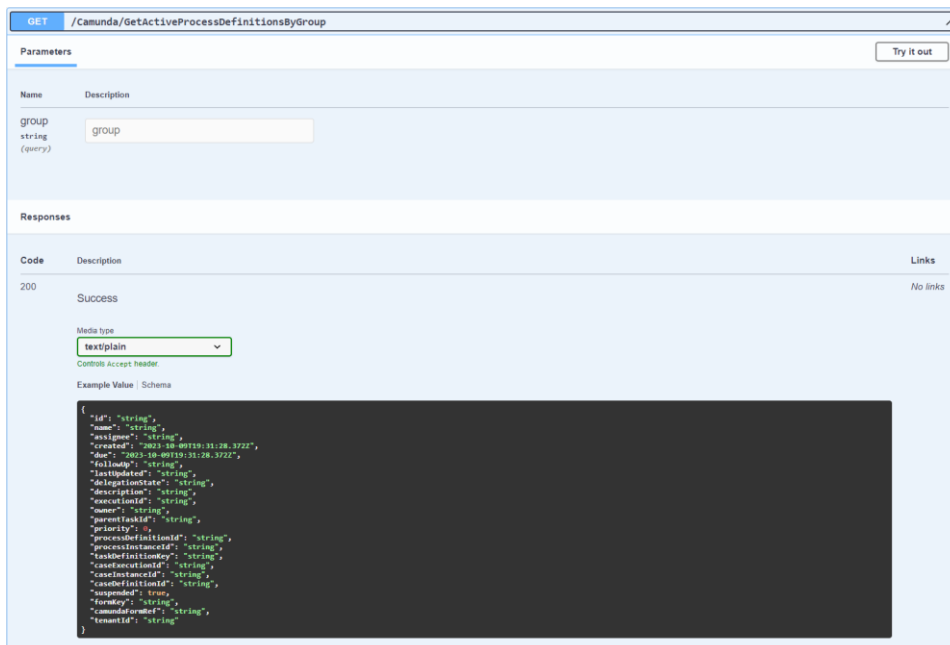
The screenshot displays the Swagger UI for the endpoint `GET /Camunda/GetProcessDefinitionsByGroup`. Under the 'Parameters' section, the 'group' parameter is defined as a string query parameter with the value 'group'. The 'Responses' section shows a 200 'Success' response with a 'text/plain' media type. The response body is a JSON object with the following structure:

```
{
  "id": "string",
  "name": "string",
  "assignee": "string",
  "created": "2023-10-09T19:31:12.529Z",
  "due": "2023-10-09T19:31:12.529Z",
  "followup": "string",
  "instanceId": "string",
  "delegationState": "string",
  "description": "string",
  "executionId": "string",
  "owner": "string",
  "parentTaskId": "string",
  "priority": 0,
  "processDefinitionId": "string",
  "processInstanceId": "string",
  "taskDefinitionId": "string",
  "taskExecutionId": "string",
  "caseExecutionId": "string",
  "caseInstance": "string",
  "caseDefinitionId": "string",
  "suspended": true,
  "formKey": "string",
  "camundaFormDef": "string",
  "tenantId": "string"
}
```

Εικόνα 25 Απεικόνιση Οθόνης Κώδικα με τον "Ορισμό διαδικασιών μιας συγκεκριμένης ομάδας"

## ➤ GetActiveProcessDefinitionsByGroup

Λαμβάνει τους ενεργούς ορισμούς διαδικασίας για μια συγκεκριμένη ομάδα.

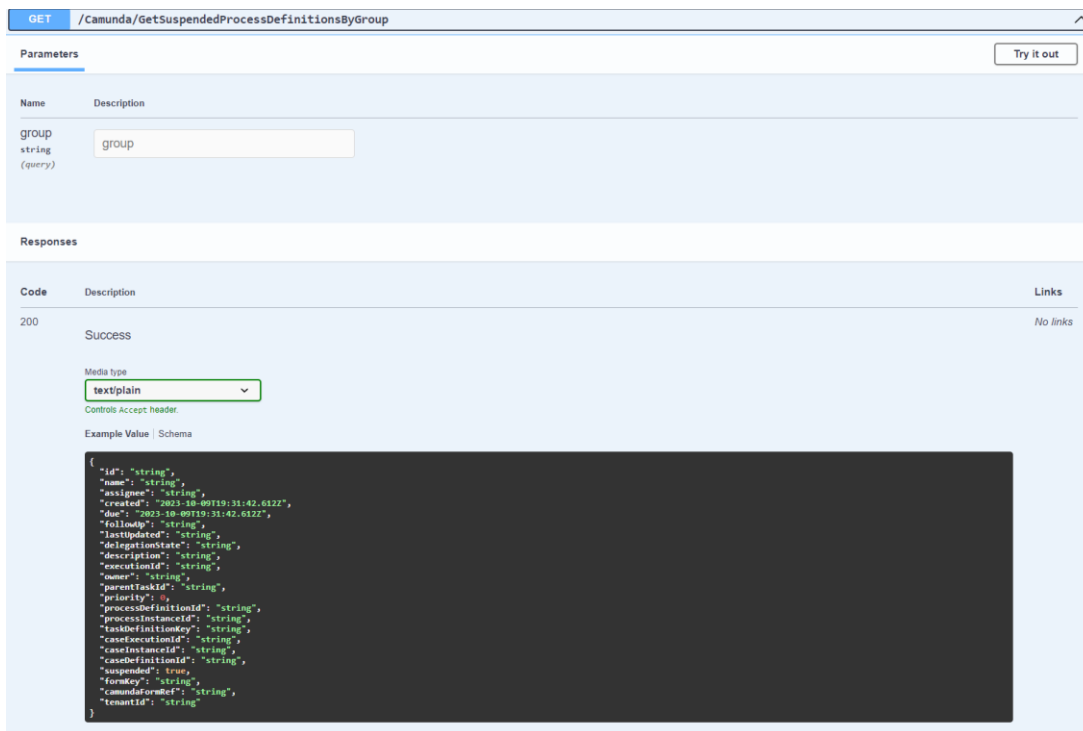


The screenshot displays the Swagger UI for the endpoint `GET /Camunda/GetActiveProcessDefinitionsByGroup`. Under the 'Parameters' section, the 'group' parameter is defined as a string query parameter with the value 'group'. The 'Responses' section shows a 200 'Success' response with a 'text/plain' media type. The response body is a JSON object with the following structure:

```
{
  "id": "string",
  "name": "string",
  "assignee": "string",
  "created": "2023-10-09T19:31:28.372Z",
  "due": "2023-10-09T19:31:28.372Z",
  "followup": "string",
  "instanceId": "string",
  "delegationState": "string",
  "description": "string",
  "executionId": "string",
  "owner": "string",
  "parentTaskId": "string",
  "priority": 0,
  "processDefinitionId": "string",
  "processInstanceId": "string",
  "taskDefinitionId": "string",
  "taskExecutionId": "string",
  "caseExecutionId": "string",
  "caseInstance": "string",
  "caseDefinitionId": "string",
  "suspended": true,
  "formKey": "string",
  "camundaFormDef": "string",
  "tenantId": "string"
}
```

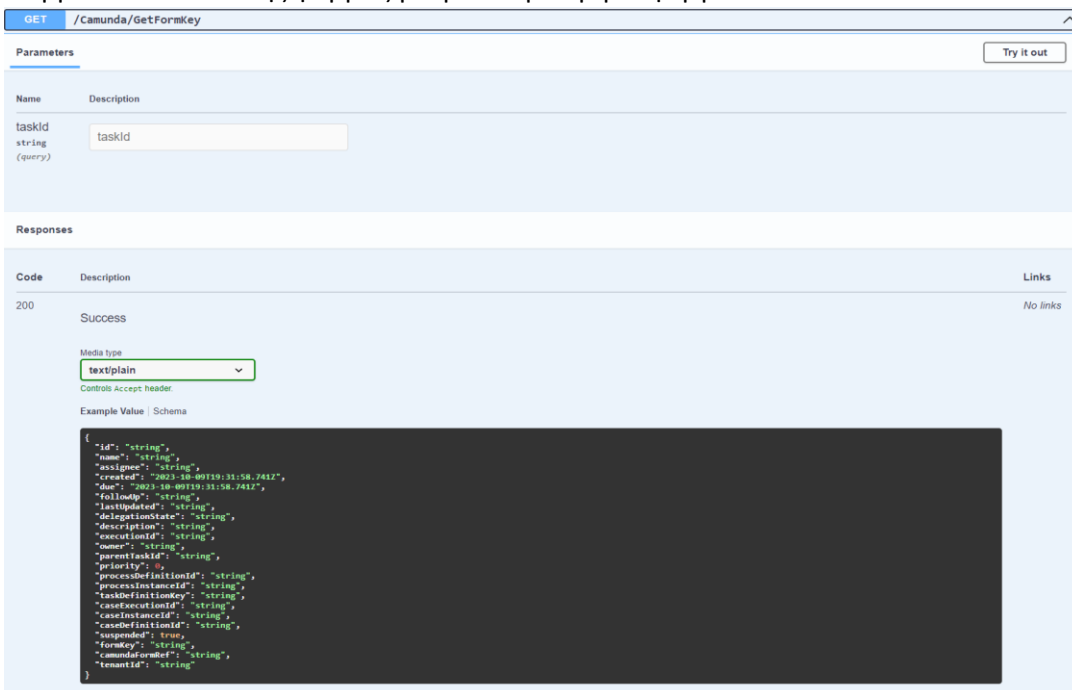
Εικόνα 26 Απεικόνιση Οθόνης Κώδικα με "Ενεργούς Ορισμούς Διαδικασίας για Συγκεκριμένη Ομάδα"

- **GetSuspendedProcessDefinitionsByGroup**  
Λαμβάνει τους ανενεργούς ορισμούς διαδικασίας για μια συγκεκριμένη ομάδα.



Εικόνα 27 Απεικόνιση Οθόνης Κώδικα με "Ανενεργούς Ορισμούς Διαδικασίας για Συγκεκριμένη Ομάδα"

- **GetFormKey**  
Λαμβάνει το κλειδί της φόρμας για μια συγκεκριμένη εργασία.



Εικόνα 28 Απεικόνιση Οθόνης Κώδικα με το "Κλειδί της φόρμας για μια Συγκεκριμένη Εργασία"

- **GetUserCompletedInstances**

Λαμβάνει τις ολοκληρωμένες περιπτώσεις διαδικασίας για έναν συγκεκριμένο χρήστη.



GET /Camunda/GetUserCompletedInstances

Parameters

Name	Description
userid string (query)	userid

Responses

Code	Description	Links
200	Success	No links

Media type: text/plain

Example Value | Schema

```
{
  "id": "string",
  "name": "string",
  "assignee": "string",
  "created": "2023-10-09T19:32:11.012Z",
  "due": "2023-10-09T19:32:11.012Z",
  "followup": "string",
  "lastUpdated": "string",
  "delegationState": "string",
  "description": "string",
  "executionId": "string",
  "name": "string",
  "parentTaskId": "string",
  "priority": 0,
  "processDefinitionId": "string",
  "processInstanceId": "string",
  "taskDefinitionKey": "string",
  "taskExecutionId": "string",
  "caseInstanceId": "string",
  "caseDefinitionId": "string",
  "suspended": true,
  "formKey": "string",
  "camundaFormRef": "string",
  "tenantId": "string"
}
```

Εικόνα 29 Απεικόνιση Οθόνης Κώδικα με "Διαδικασίες για συγκεκριμένο χρήστη"

### ➤ GetUserActiveInstances

Λαμβάνει τις ενεργές περιπτώσεις διαδικασίας για έναν συγκεκριμένο χρήστη.

GET /Camunda/GetUserActiveInstances

Parameters

Name	Description
userid string (query)	userid

Responses

Code	Description	Links
200	Success	No links

Media type: text/plain

Example Value | Schema

```
{
  "id": "string",
  "name": "string",
  "assignee": "string",
  "created": "2023-10-09T19:32:23.274Z",
  "due": "2023-10-09T19:32:23.274Z",
  "followup": "string",
  "lastUpdated": "string",
  "delegationState": "string",
  "description": "string",
  "executionId": "string",
  "name": "string",
  "parentTaskId": "string",
  "priority": 0,
  "processDefinitionId": "string",
  "processInstanceId": "string",
  "taskDefinitionKey": "string",
  "taskExecutionId": "string",
  "caseInstanceId": "string",
  "caseDefinitionId": "string",
  "suspended": true,
  "formKey": "string",
  "camundaFormRef": "string",
  "tenantId": "string"
}
```

Εικόνα 30 Απεικόνιση Οθόνης Κώδικα με "Ενεργές Διαδικασίες για συγκεκριμένο χρήστη"

### ➤ GetTaskVariables

Λαμβάνει τις μεταβλητές μιας εργασίας.

GET /Camunda/task/{taskId}/variables

Parameters Try it out

Name	Description
taskId * required string (path)	<input type="text" value="taskId"/>

Responses

Code	Description	Links
200	Success	No links

Media type:

Controls Accept header:

Example Value | Schema

```
{
  "additionalProp1": "string",
  "additionalProp2": "string",
  "additionalProp3": "string"
}
```

Εικόνα 31 Απεικόνιση Οθόνης Κώδικα με "Μεταβλητές μια εργασίας"

### ➤ **GetCompletedTaskVariables**

Λαμβάνει τις ολοκληρωμένες μεταβλητές για μια περίπτωση διαδικασίας.

GET /Camunda/task/completedVariables

Parameters Try it out

Name	Description
processInstanceId string (query)	<input type="text" value="processInstanceId"/>

Responses

Code	Description	Links
200	Success	No links

Media type:

Controls Accept header:

Example Value | Schema

```
{
  "additionalProp1": "string",
  "additionalProp2": "string",
  "additionalProp3": "string"
}
```

Εικόνα 32 Απεικόνιση Οθόνης Κώδικα με "Ολοκληρωμένες Μεταβλητές"

### ➤ **GetActiveTaskVariables**

Λαμβάνει τις ενεργές μεταβλητές για μια περίπτωση διαδικασίας.

GET /Camunda/task/activeVariables

Parameters Try it out

Name	Description
processInstanceid string (query)	<input type="text" value="processInstanceid"/>

Responses

Code	Description	Links
200	Success	No links

Media type:

Controls: Accept header

Example Value | Schema

```
{
  "additionalProp1": "string",
  "additionalProp2": "string",
  "additionalProp3": "string"
}
```

Εικόνα 33 Απεικόνιση Οθόνης Κώδικα με "Ενεργές μεταβλητές"

## HTTP PUT Endpoints:

**SuspendProcess:** Αναστέλλει μια διαδικασία βάσει του αναγνωριστικού της διαδικασίας. Απαιτεί την παράμετρο **processId**, που είναι το αναγνωριστικό της διαδικασίας που θα αναστείλει.

PUT /Camunda/SuspendProcess

Parameters Try it out

Name	Description
processId string (query)	<input type="text" value="processId"/>

Responses

Code	Description	Links
200	Success	No links

Εικόνα 34 Απεικόνιση Οθόνης Κώδικα "Suspend Process"

**ActivateProcess:** Ενεργοποιεί μια διαδικασία που είναι ανεσταλμένη, βάσει του αναγνωριστικού της διαδικασίας. Απαιτεί την παράμετρο **processId**, που είναι το αναγνωριστικό της διαδικασίας που θα ενεργοποιήσετε.

PUT /Camunda/ActivateProcess

Parameters Try it out

Name	Description
processId string (query)	<input type="text" value="processId"/>

Responses

Code	Description	Links
200	Success	No links

Εικόνα 35 Απεικόνιση Οθόνης Κώδικα "Activate Process"

## HTTP POST Endpoints:

**ClaimTask:** Κατοχυρώνει μια εργασία σε έναν συγκεκριμένο χρήστη. Απαιτεί τις παραμέτρους **taskId** (το αναγνωριστικό της εργασίας) και **userId** (το αναγνωριστικό του χρήστη στον οποίο θα γίνει η κατοχύρωση).

POST /Camunda/tasks/{taskId}/Claim

Parameters Try it out

Name	Description
<b>taskId</b> * required string (path)	<input type="text" value="taskId"/>
userId string (query)	<input type="text" value="userId"/>

Responses

Code	Description	Links
200	Success	No links

Εικόνα 36 Απεικόνιση Οθόνης Κώδικα "Claim Task"

**UnClaimTask:** Αφαιρεί την κατοχύρωση μιας εργασίας από τον χρήστη που την έχει κατοχυρώσει. Απαιτεί την παράμετρο **taskId** (το αναγνωριστικό της εργασίας).



POST /Camunda/tasks/{taskId}/UnClaim

Parameters

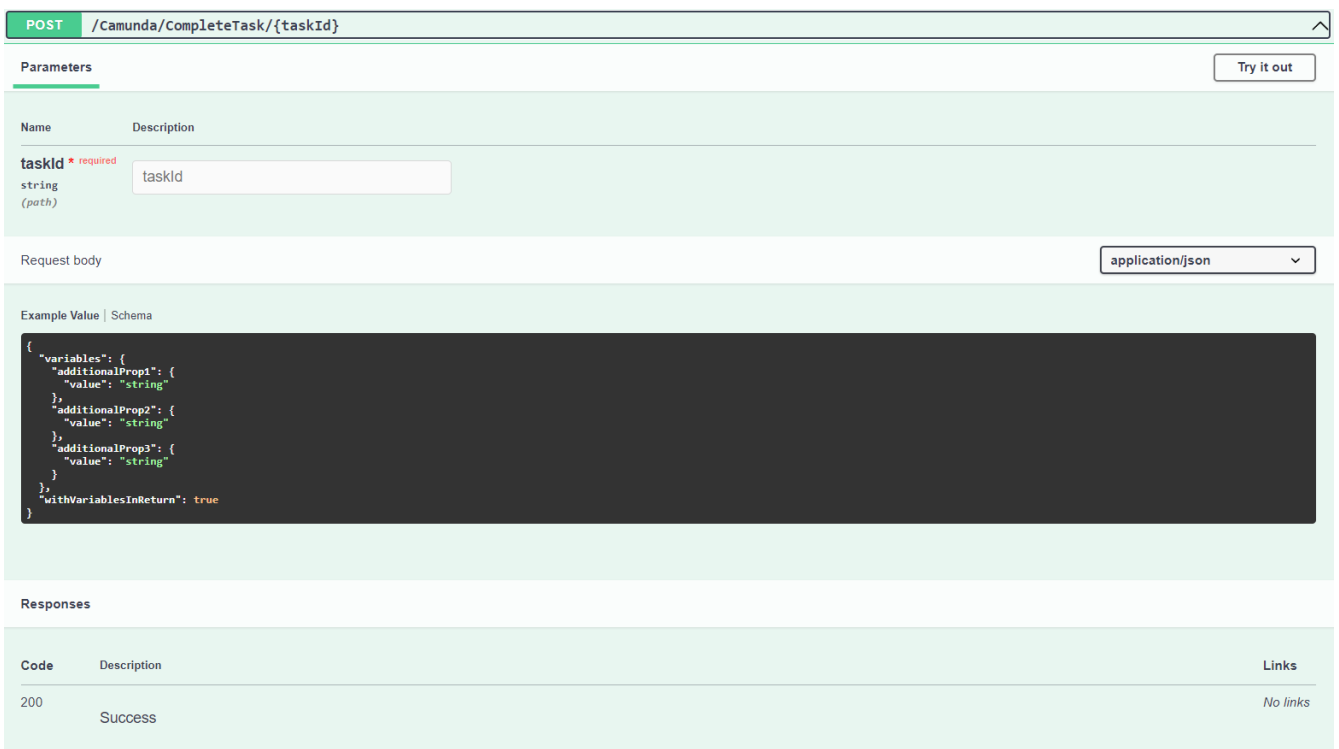
Name	Description
<b>taskId</b> * required string (path)	taskId

Responses

Code	Description	Links
200	Success	No links

Εικόνα 37 Απεικόνιση Οθόνης Κώδικα "Unclaim Task"

**CompleteTask:** Ολοκληρώνει μια εργασία και παρέχει μεταβλητές κατά την ολοκλήρωση. Απαιτεί την παράμετρο **taskId** (το αναγνωριστικό της εργασίας) και δέχεται ένα αντικείμενο **CompleteRequest** ως σώμα του αιτήματος HTTP. Το αντικείμενο **CompleteRequest** περιέχει μεταβλητές που θα προστεθούν κατά την ολοκλήρωση της εργασίας.



POST /Camunda/CompleteTask/{taskId}

Parameters

Name	Description
<b>taskId</b> * required string (path)	taskId

Request body

application/json

Example Value | Schema

```
{
  "variables": {
    "additionalProp1": {
      "value": "string"
    },
    "additionalProp2": {
      "value": "string"
    },
    "additionalProp3": {
      "value": "string"
    }
  },
  "withVariablesInReturn": true
}
```

Responses

Code	Description	Links
200	Success	No links

Εικόνα 38 Απεικόνιση Οθόνης Κώδικα "Complete Task"

**StartProcess:** Ξεκινά μια νέα διαδικασία βάσει του αναγνωριστικού της διαδικασίας και παρέχει παραμέτρους εκκίνησης. Απαιτεί την παράμετρο **processId** (το αναγνωριστικό της

διαδικασίας) και δέχεται ένα αντικείμενο **StartRequest** ως σώμα του αιτήματος HTTP. Το αντικείμενο **StartRequest** περιλαμβάνει παραμέτρους εκκίνησης της διαδικασίας.

**POST** /Camunda/StartInstance/{processId}

Parameters Try it out

Name	Description
<b>processId</b> * required string (path)	processId

Request body application/json

Example Value | Schema

```
{
  "variables": {
    "protocol": {
      "value": "string",
      "type": "string"
    },
    "citizenUsername": {
      "value": "string",
      "type": "string"
    }
  },
  "businessKey": "string"
}
```

Responses

Code	Description	Links
200	Success	No links

Εικόνα 39 Απεικόνιση Οθόνης Κώδικα "Start Process"

**Ελεγκτής GroupsController:** Αυτή είναι η κύρια κλάση του ελεγκτή που είναι υπεύθυνη για τη διαχείριση των ομάδων.

**ApplicationDbContext:** Η κλάση **ApplicationDbContext** αντιπροσωπεύει τον συνδεδεμένο προνομιούχο προορισμό δεδομένων (database context) και χρησιμοποιείται για την αλληλεπίδραση με τη βάση δεδομένων.

**GET: api/Groups:** Αυτή είναι μια HTTP GET μέθοδος που επιστρέφει όλες τις ομάδες από τη βάση δεδομένων. Χρησιμοποιεί το Entity Framework για να ανακτήσει τις ομάδες από τη βάση δεδομένων.

The screenshot displays a REST client interface for the endpoint `GET /api/Groups`. The 'Parameters' section is empty, indicating no parameters are required. The 'Responses' section shows a table with one entry: a 200 status code and a 'Success' description. Below the table, the 'Media type' is set to 'text/plain'. An 'Example Value' section shows a JSON array: `[ { 'id': 0, 'name': 'string' } ]`.

Εικόνα 40 Απεικόνιση Οθόνης Κώδικα "Get Groups"

**POST: api/Groups:** Αυτή είναι μια HTTP POST μέθοδος που προσθέτει μια νέα ομάδα στη βάση δεδομένων. Προσθέτει τη νέα ομάδα στο "ApplicationDbContext.Groups". Χρησιμοποιεί το Entity Framework για να αποθηκεύσει τη νέα ομάδα στη βάση δεδομένων. Επιστρέφει τη δημιουργημένη ομάδα μαζί με τον κωδικό κατάστασης HTTP 201 Created.

The screenshot shows a REST client interface for a POST request to the endpoint `/api/Groups`. The request body is a JSON object: `{ "id": 0, "name": "string" }`. The response is a 200 status code with a description of "Success" and the same JSON object in the body. The media type is set to `text/plain`.

Εικόνα 41 Απεικόνιση Οθόνης Κώδικα "Post Groups"

### Ελεγκτής «UsersController»:

Αυτός ο ελεγκτής χρησιμοποιείται για τη διαχείριση των χρηστών στη βάση δεδομένων.

Ο κατασκευαστής του ελεγκτή δέχεται ένα αντικείμενο `ApplicationDbContext` ως παράμετρο και το αποθηκεύει για περαιτέρω χρήση.

Η μέθοδος HTTP "GET" "GetUsers" ανακτά όλους τους χρήστες από τη βάση δεδομένων και τους επιστρέφει σε μια απόκριση HTTP με κωδικό κατάστασης 200 (OK).

Η μέθοδος HTTP "POST" "CreateUser" δημιουργεί ένα νέο χρήστη στη βάση δεδομένων με βάση τα δεδομένα που παρέχονται στο σώμα του αιτήματος HTTP. Στη συνέχεια, επιστρέφει το νέο χρήστη σε μια απόκριση HTTP με κωδικό κατάστασης 200 (OK).

Τέλος δημιουργήθηκε η κλάση `Program.cs` όπου ο κώδικας περιλαμβάνει τη ρύθμιση και τη διαμόρφωση μιας εφαρμογής ASP.NET Core. Ας αναλύσουμε τον κώδικα ανά τμήματα:

- **Εγκατάσταση υπηρεσιών:** Ο κώδικας ξεκινά με τη δημιουργία ενός νέου builder για την εφαρμογή ASP.NET Core και την εγκατάσταση υπηρεσιών στον container του. Οι υπηρεσίες που εγκαθίστανται είναι οι εξής:
- **AddControllers():** Εγκαθίσταται η υπηρεσία για τους controllers της εφαρμογής.



- **AddEndpointsApiExplorer():** Εγκαθίσταται η υπηρεσία που επιτρέπει την ανίχνευση των επικαλούμενων σημείων (endpoints) στην εφαρμογή.
- **AddSwaggerGen():** Εγκαθίσταται η υπηρεσία για τη δημιουργία της τεκμηρίωσης Swagger/OpenAPI.
- **AddSingleton():** Εγκαθίσταται μια μοναδική παρουσία της υπηρεσίας CamundaService, η οποία παρέχεται ως singleton.
- **Ρύθμιση CORS (Cross-Origin Resource Sharing):** Στο τμήμα αυτό, ορίζεται μια πολιτική CORS με το όνομα "corspolicy" που επιτρέπει αιτήσεις από τη διεύθυνση "http://localhost:3000" με οποιαδήποτε μέθοδο (method) και κεφαλίδα (header).
- **Ρύθμιση της σύνδεσης με τη βάση δεδομένων:** Στο τμήμα αυτό, ορίζεται η σύνδεση με τη βάση δεδομένων μέσω του AddDbContext. Η σύνδεση επιλέγεται από τις ρυθμίσεις διαμόρφωσης (configuration) της εφαρμογής, όπου αναζητείται η συμβολοσειρά σύνδεσης με το όνομα "DefaultConnection".
- **Ρύθμιση Swagger/OpenAPI:** Στο τμήμα αυτό, ορίζεται η δημιουργία της τεκμηρίωσης Swagger/OpenAPI με τον τίτλο "My API" και την έκδοση "v1". Επιπλέον, εμφανίζεται η διεπαφή Swagger UI για την περιήγηση και τον έλεγχο των διαθέσιμων σημείων (endpoints) της εφαρμογής.
- **Ρύθμιση περιβάλλοντος ανάπτυξης:** Στο τμήμα αυτό, γίνεται έλεγχος αν η εφαρμογή εκτελείται σε περιβάλλον ανάπτυξης. Αν ναι, ενεργοποιείται η διεπαφή Swagger UI για την αναπαράσταση της τεκμηρίωσης Swagger/OpenAPI.
- **Ρύθμιση των middleware:** Στο τμήμα αυτό, πραγματοποιείται η ρύθμιση των middleware που επεξεργάζονται τις αιτήσεις. Συγκεκριμένα, γίνεται χρήση του UseCors για τη διαχείριση των πολιτικών CORS, του UseHttpsRedirection για την ανακατεύθυνση των HTTP αιτήσεων σε HTTPS, του UseSwagger για την παροχή της τεκμηρίωσης Swagger/OpenAPI και του UseSwaggerUI για την παρουσίαση της διεπαφής Swagger UI. Επιπλέον, χρησιμοποιείται το UseAuthorization για την ενεργοποίηση του μεσολαβητή εξουσιοδότησης (authorization middleware) για την πιστοποίηση των αιτήσεων.
- **Εκτέλεση της εφαρμογής:** Στο τμήμα αυτό, ορίζεται η εκτέλεση της εφαρμογής, καθώς και η αντιστοίχιση (mapping) των controllers για την ανταπόκριση στις αιτήσεις που λαμβάνονται.

Συνοπτικά, ο παραπάνω κώδικας περιγράφει την ρύθμιση και τη διαμόρφωση μιας εφαρμογής ASP.NET Core, συμπεριλαμβανομένης της εγκατάστασης των απαραίτητων υπηρεσιών, της διαχείρισης των αιτήσεων, της επικοινωνίας με τη βάση δεδομένων και της παροχής της τεκμηρίωσης Swagger/OpenAPI.

## Αποτελέσματα και αξιολόγηση

Η ενσωμάτωση και η χρήση του Camunda στο πλαίσιο των εργασιών αποδείχθηκε υψίστης σημασίας και προώθησε βαθμιαία την απόδοση και την αποτελεσματικότητα των διαφόρων στοιχείων που συμμετείχαν στον οργανισμό.

Αυτή η σημαντική επένδυση στην τεχνολογία διευκόλυνε τη διαχείριση αιτήσεων με εξαιρετική άνεση και ασύγκριτη ταχύτητα, υπερβαίνοντας σημαντικά τις αποτελεσματικότητες των προηγούμενων μεθόδων.

Ένα από τα κύρια πλεονεκτήματα της εφαρμογής του Camunda ήταν η δυνατότητα παρακολούθησης της προόδου των αιτήσεων που δοκιμάστηκαν σε πραγματικό χρόνο. Αυτό σημαίνει ότι υπήρξε η δυνατότητα να ελεγχθεί και να παρακολουθηθεί η διαδικασία από την αρχή έως το τέλος, εξασφαλίζοντας ότι κάθε βήμα εκτελείται με ακρίβεια και χωρίς καθυστερήσεις. Αυτή η δυνατότητα παρακολούθησης σε πραγματικό χρόνο επέτρεψε την άμεση ανίχνευση και διόρθωση οποιωνδήποτε προβλημάτων ή λαθών προκύψουν κατά τη διάρκεια της διαδικασίας. Συνολικά, η ενσωμάτωση του Camunda στο οικοσύστημα των εργασιών είχε θετική επίδραση πέραν των προσδοκιών. Βελτιώθηκε σημαντικά η απόδοση και η αποτελεσματικότητα του οργανισμού, διευκολύνοντας τη ροή εργασιών και τη διαχείριση αιτήσεων. Οι εργαζόμενοι αισθάνθηκαν τη διαφορά αμέσως και είδαν την αναβάθμιση της αποδοτικότητας και της ακρίβειας των διαδικασιών εργασίας τους.

Επιπρόσθετα, η ευελιξία του Camunda επέτρεψε την προσαρμογή των διαδικασιών στις μεταβαλλόμενες ανάγκες του οργανισμού. Μπορούσαμε να προσθέτουμε, να τροποποιούμε και να εξατομικεύουμε διαδικασίες με ευκολία, εξασφαλίζοντας ότι η εργασία παρέμενε πάντα συμβατή με τις αλλαγές στο περιβάλλον.

Ως εκ τούτου, η εκτεταμένη τεκμηρίωση και η εκπαίδευση που παρέχονται από την ομάδα υποστήριξης του Camunda βοήθησαν να αξιοποιήσουμε στο έπακρο τις δυνατότητες του συστήματος. Οι εκπαιδευτικοί πόροι και οι συμβουλές από ειδικούς συνέβαλαν στην εξειδίκευσή και στη βελτιστοποίηση της χρήσης του Camunda. Σε συνδυασμό με την ανώτερη τεχνική υποστήριξη που προσφέρει το Camunda, αισθανόμαστε πλέον πιο εξειδικευμένοι και έτοιμοι να αντιμετωπίσουμε πολύπλοκες διαδικασίες εργασίας με αυξημένη αυτοπεποίθηση.

Συνολικά, η χρήση του Camunda σηματοδότησε μια νέα εποχή στην εργασία μας, επιτρέποντάς μας να επιτύχουμε αποτελεσματικότερες και αποδοτικότερες διαδικασίες εργασίας. Αποτελεί ένα κρίσιμο εργαλείο για τον σύγχρονο οργανισμό και αναμφίβολα θα συνεχίσει να αποτελεί τον πυρήνα της προσπάθειάς μας για συνεχή βελτίωση και ανάπτυξη.

### Αξιολόγηση αποτελεσμάτων

Η αξιοποίηση ενός προηγμένου συστήματος αυτοματοποίησης διαδικασιών, όπως το Camunda, ανέκαθεν αποτελούσε κρίσιμο στοιχείο για την διαχείριση και επεξεργασία πιστοποιητικών από τους πολίτες ενός δήμου. Η διπλωματική εργασία μας επικεντρώθηκε στη χρήση αυτής της πλατφόρμας για τη δημιουργία ενός πρωτοποριακού συστήματος διαχείρισης και έκδοσης πιστοποιητικών, με στόχο την

απλοποίηση των διαδικασιών, τη μείωση της πολυπλοκότητας της προγραμματιστικής εργασίας και τη διευκόλυνση της ευέλικτης προσαρμογής των αιτημάτων χωρίς την ανάγκη συνεχούς επεξεργασίας του κώδικα. Στόχοι που περιλαμβάνονταν στην έρευνά μας περιλάμβαναν τη βελτιστοποίηση της διαχείρισης διαδικασιών, τη μείωση της πολύπλοκης προγραμματιστικής εργασίας που απαιτείται για την υλοποίηση αυτών των διαδικασιών και τη διασφάλιση της δυνατότητας εύκολης προσαρμογής των αιτημάτων μας χωρίς την ανάγκη συνεχούς επεξεργασίας του κώδικα της εφαρμογής.

Η εφαρμογή του Camunda στην πράξη αποδείχθηκε εξαιρετικά αποδοτική και πρακτική. Ένα από τα σημαντικότερα πλεονεκτήματα είναι η δι-επαφή σχεδίασης του Camunda, που παρέχει μια οπτική αναπαράσταση των διαδικασιών. Αυτό επιτρέπει σε εμάς, ως σχεδιαστές διαδικασιών, να καθορίσουμε τις απαιτούμενες αλληλεπιδράσεις και τη ροή των διαδικασιών με μεγάλη ευκολία και λεπτομέρεια.

Αυτή η προσέγγιση επιτρέπει στους προγραμματιστές να επικεντρωθούν στη βελτιστοποίηση και εξέλιξη των διαδικασιών χωρίς την ανάγκη για συνεχείς αλλαγές στον πηγαίο κώδικα της εφαρμογής. Αντί αυτού, δύναται να προσαρμοστούν τα αιτήματα και τις διαδικασίες μέσω της δι-επαφής σχεδίασης (modeler), εστιάζοντας στην προσθήκη ή αφαίρεση τμημάτων από τις διαδικασίες και τις αλληλεπιδράσεις.

Η αξιολόγηση των αποτελεσμάτων απέδειξε ότι η εφαρμογή του Camunda επέτυχε τους προαναφερθέντες στόχους της αυτοματοποίησης διαδικασιών για την έκδοση πιστοποιητικών. Η ανάπτυξη και βελτιστοποίηση των διαδικασιών μπορεί να επιτευχθεί, ενώ οι αλλαγές στα αιτήματα μπορούν να γίνουν εύκολα μέσω της δι-επαφής σχεδίασης από οποιονδήποτε εμπλεκόμενο. Τέλος, η αυτοματοποίηση διαδικασιών μέσω του Camunda επέτρεψε την αποτελεσματική έκδοση πιστοποιητικών για τους πολίτες ενός δήμου, εξοικονομώντας πολύτιμο χρόνο και πόρους μας και επιτρέποντας ευέλικτες αλλαγές στα αιτήματα χωρίς περιττές παρεμβάσεις.

Συνοψίζοντας, η χρήση του Camunda αποτελεί κρίσιμο στοιχείο για την επιτυχία μας στον τομέα της διαχείρισης και έκδοσης πιστοποιητικών από πλευράς του δημόσιου οργανισμού, εξασφαλίζοντας την αποτελεσματική και αποδοτική εκτέλεση των διαδικασιών, ενώ παράλληλα προσφέρει την αναγκαία ευελιξία για αλλαγές και βελτιώσεις χωρίς περιττή πολυπλοκότητα στον κώδικα της εφαρμογής.

## Επόμενα βήματα

Βαθύτερες βελτιώσεις στις εφαρμογές και αναβαθμίσεις του Camunda μπορούν να περιλαμβάνουν την εγκατάστασή τους σε ένα περιβάλλον Kubernetes για αυξημένη ευελιξία και απόδοση. Επίσης, είναι απαραίτητο να δημιουργηθεί και να παραμετροποιηθεί ένα σύστημα Continuous Integration/Continuous Deployment (CI/CD) σύμφωνα με τις βέλτιστες πρακτικές του Application Lifecycle Management

για την αυτοματοποίηση της διαδικασίας ανάπτυξης και αναβάθμισης των εφαρμογών. Παράλληλα, άλλα σημαντικά βήματα περιλαμβάνουν τη βελτιστοποίηση της ασφάλειας, την εφαρμογή παρακολούθησης, την υλοποίηση πρακτικών DevOps, τη διαχείριση της δομής του Camunda, τη σχεδίαση με αρχές Microservices, την εκπαίδευση του προσωπικού και τη διαχείριση αντιστάσεων. Όλα αυτά τα βήματα θα συμβάλουν στη βελτίωση της απόδοσης και της ασφάλειας των εφαρμογών σας.

## Εγκατάσταση των εφαρμογών σε Kubernetes

Για τον αυτοματισμό της ανάπτυξης, κλιμάκωσης και της διαχείρισης εφαρμογών, η Google έχει δημιουργήσει ένα σύστημα ανοιχτού κώδικα ονόματι Kubernetes (K8s). Στην πράξη, το εν λόγω σύστημα, συγκεντρώνει και ομαδοποιεί όλους τους πόρους των δοχείων (containers), που συνθέτουν μια εφαρμογή (Kubernetes, T. (2019). Kubernetes. Retrieved May, 24, 2019).

Στα χαρακτηριστικά του Kubernetes, περιλαμβάνεται η ενορχήστρωση πόρων (pods) και δοχείων (containers). Σύμφωνα με Luksa (2017), ένα pod μπορεί να αποτελείται από έναν, ή και περισσότερους Containers που μοιράζονται το ίδιο namespace και τον ίδιο όγκο αποθήκευσης. Με αυτό τον τρόπο, εξασφαλίζεται η τελεσφόρος συνεργασία μεταξύ των containers μέσα στο ίδιο pod.

Ένα άλλο χαρακτηριστικό το οποίο μπορεί να χρησιμοποιηθεί από το Kubernetes είναι το helm. Το Helm βασίζεται σε ένα σύστημα γνωστό ως "charts," τα οποία είναι πακέτα που περιλαμβάνουν όλες τις πληροφορίες και τις διαμορφώσεις που απαιτούνται για την εγκατάσταση και τη διαμόρφωση μιας εφαρμογής στο Kubernetes. Ένα chart μπορεί να περιλαμβάνει τις διάφορες υπηρεσίες, τις εξαρτήσεις, τις παραμέτρους, και τις διαμορφώσεις που απαιτούνται για τη σωστή ανάπτυξη και λειτουργία της εφαρμογής. Με το Helm, οι προγραμματιστές και οι διαχειριστές μπορούν να απλοποιήσουν τη διαδικασία ανάπτυξης, διαχείρισης και ενημέρωσης των εφαρμογών τους στο Kubernetes, επιτρέποντάς τους να επαναχρησιμοποιούν charts και να αποφεύγουν την ανάγκη να δημιουργούν την υποδομή και τις διαμορφώσεις από την αρχή κάθε φορά που αναπτύσσουν μια εφαρμογή. Αυτό καθιστά το Helm ένα πολύτιμο εργαλείο για την αυτοματοποίηση και τη διαχείριση εφαρμογών στο Kubernetes.

Επίσης, μια άλλη χαρακτηριστική λειτουργία του Kubernetes, είναι η ανακάλυψη υπηρεσιών και πόρων δικτύων. Όπως αναφέρει το Kubernetes (2021), το σύστημα παρέχει ένα ισχυρό μοντέλο δικτύωσης για την αποτελεσματική επικοινωνία μεταξύ των πόρων και των υπηρεσιών. Οι υπηρεσίες λειτουργούν ως εξισορροπιστές φόρτου, εκθέτοντας τους πόρους σε άλλους πόρους ή σε εξωτερικούς χρήστες. Μέσω της ανάπτυξης συγκεκριμένων πολιτικών δικτύου, ενδυναμώνεται η ασφάλεια του cluster και κατά συνέπεια, γίνεται πιο εύκολα η απομόνωση του φόρτου.

Ακόμη, το Kubernetes έχει την δυνατότητα να αναπτύξει ή να τροποποιήσει τους πόρους των δοχείων που συνθέτουν μια εφαρμογή. «Μερικιά στην

αποτελεσματική κλιμάκωση των πανομοιότυπων πόρων, την ανάπτυξη ενημερωμένου κώδικα με ελεγχόμενο τρόπο, ή την επιστροφή σε προηγούμενη έκδοση εάν είναι αυτό απαραίτητο».

Επιπλέον, το Kubernetes «εισάγει προηγμένες έννοιες για να επεκτείνει τις δυνατότητές του. Οι πόροι εισόδου επιτρέπουν τη δρομολόγηση εξωτερικής πρόσβασης, παρέχοντας ακριβή έλεγχο της κυκλοφορίας HTTP και HTTPS. Εν συνεχεία, τα namespace διευκολύνουν την οργάνωση και την απομόνωση των πόρων, απλοποιώντας τη διαχείριση πολλαπλών εφαρμογών σε ένα ενιαίο cluster»(Kubernetes, 2021).

Γνωρίζοντας, ήδη, ότι το Camunda BPM είναι μια πλατφόρμα ανοιχτού κώδικα για τη ροή των εργασιών και την αυτοματοποίηση των επιχειρησιακών αποφάσεων, θα αναλυθεί παρακάτω, ο τρόπος εκτέλεσης της πλατφόρμας Camunda στο περιβάλλον του Kubernetes cluster.

Προαιρετικό, αλλά καθόλου ασήμαντο βήμα είναι η δημιουργία ενός διαμορφωμένου χώρου (namespace) για το Camunda. Για το σκοπό αυτό, εκτελείται η εντολή «**kubectl create namespace camunda**». Η απομόνωση του Camunda, που προσφέρει το Kubernetes cluster, μεριμνά στην καλύτερη διαχείριση των διαθέσιμων πόρων του.

Κατόπιν, πραγματοποιείται η προσθήκη Camunda στα Helm διαγράμματα «Charts». Για την πρόσβαση στα διαγράμματα Helm του Camunda, χρειάζεται να προστεθεί το αποθετήριο του Helm κατά την φάση της διαμόρφωσης του τελευταίου. Για αυτό το σκοπό, εκτελείται η εντολή «**helm repo add camunda**». Κατόπιν αυτού εκτελώντας την εντολή «**helm install camunda camunda/camunda-bpm -n camunda**» ξεκινά η εγκατάσταση της πλατφόρμας στο Kubernetes.

Είναι σημαντικό να αναφερθεί ότι, κατά την διάρκεια διαμόρφωσης του νέου αποθετηρίου Helm, παραμένουν ενημερωμένα τα ήδη υπάρχοντα (αποθετήρια). Η ενημέρωση παρέχει την πρόσβαση σε πιο πρόσφατες εκδόσεις helm Charts. Ως εκ τούτου, μπορεί να εκτελεστεί η εντολή «**helm repo update**»

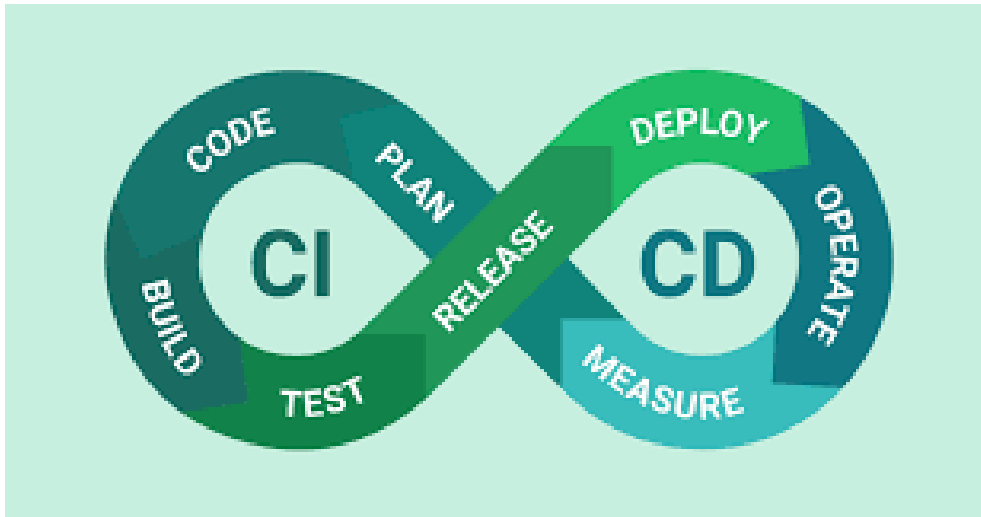
Το Camunda δίνει την ευκαιρία να διαμορφωθούν τα εκάστοτε μέρη του. Ειδικότερα, δημιουργείται ένα αρχείο παραμέτρων και μέσα σε εκείνο, καθορίζονται οι επιλογές διαμόρφωσης της εφαρμογής όπως είναι για παράδειγμα οι ρυθμίσεις βάσης δεδομένων, διευθύνσεις εξωτερικών συνδέσεων και διάθεση των πόρων. Οι τιμές προσαρμόζονται σύμφωνα με τις απαιτήσεις του πελάτη και προκαθορίζονται πριν την εγκατάσταση του Camunda.

Εφόσον, ακολουθηθεί η παραπάνω διαδικασία, το Camunda είναι έτοιμο να εκτελεστεί στο περιβάλλον του Kubernetes cluster και να είναι προσβάσιμο στην διαδικτυακή διεύθυνση που έχουν οριστεί τις παραμέτρους κατά την εγκατάσταση.

## CI/CD

Η Συνεχής Ενσωμάτωση(Continues Integration) και Συνεχής Παράδοση (Continues Delivery) εφεξής CI/CD, είναι ζωτικές πρακτικές στον τομέα της ανάπτυξης

λογισμικού, σχεδιασμένες για τη βελτίωση της ανάπτυξης, δοκιμής και στις αναπτύξεις εφαρμογών. Ο στόχος του CI/CD είναι η επιτάχυνση και η αυτοματοποίηση του κύκλου ανάπτυξης λογισμικού, διασφαλίζοντας ότι οι αλλαγές στον κώδικα ενσωματώνονται, δοκιμάζονται και αναπτύσσονται αποτελεσματικά.



Εικόνα 42 Απεικόνιση CI/CD

Η Συνεχής Ενσωμάτωση (CI) περιλαμβάνει την τακτική και αυτοματοποιημένη ενσωμάτωση των αλλαγών στον κώδικα από πολλούς προγραμματιστές σε ένα κοινό αποθετήριο. Αυτόματες διαδικασίες εκκινούν την δημιουργία νέας έκδοσης της εφαρμογής καθώς και δοκιμαστικά σενάρια που ενεργοποιούνται κάθε φορά που γίνονται αλλαγές στον κώδικα. Αυτή η διαδικασία βοηθά στον εντοπισμό προβλημάτων κατά την ενσωμάτωση και προβληματικών καταστάσεων του κώδικα, μειώνοντας την πιθανότητα εμφάνισης δαπανηρών προβλημάτων (π.χ. ο χρόνος για την επίλυση τους) κατά την ανάπτυξη των εφαρμογών (Duvall κ.ά., 2007).

Επίσης, η Συνεχής Παράδοση (CD) επεκτείνει την Συνεχής Ενσωμάτωση (CI) με την αυτοματοποίηση της ανάπτυξης των αλλαγών στον κώδικα από το περιβάλλον της ανάπτυξης μέχρι το περιβάλλον της παραγωγής. Οι αυτοματοποιημένες διαδικασίες παράδοσης εξασφαλίζουν μια συνεπή και απαλλαγμένη από σφάλματα διαδικασία. Αυτή η πρακτική μειώνει τον κίνδυνο που συνδέεται με τις χειροκίνητες παραδόσεις και επιτρέπει την γρήγορη και αξιόπιστη ενημέρωση των εφαρμογών (Humble & Farley, 2010)

Συνοψίζοντας, το CI/CD επιταχύνει την ανάπτυξη, βελτιώνει την ποιότητα του κώδικα, προωθεί τη συνεργασία και παρέχει γρήγορα σχόλια και αποτελέσματα στους προγραμματιστές. Αυτές οι πρακτικές έχουν γίνει ουσιώδεις στη σύγχρονη ανάπτυξη εφαρμογών, επιτρέποντας στις ομάδες να παράγουν υψηλής ποιότητας λογισμικά πιο αστην αγορά υπάρχουν πολλά εργαλεία που προσφέρουν λύσεις για την αυτοματοποίηση και την αποτελεσματική διαχείριση των διαδικασιών

ανάπτυξης και παράδοσης λογισμικού (CI/CD). Μερικά από αυτά είναι τα ακόλουθα:

**Jenkins:** Το Jenkins είναι ένα εργαλείο αυτοματισμού που χρησιμοποιείται συχνά για τη διαχείριση της συνεχούς ενσωμάτωσης (Continuous Integration - CI). Είναι γραμμένο σε Java και διαθέτει πρόσθετα που είναι σχεδιασμένα ειδικά για την εκτέλεση διαδικασιών CI. Το Jenkins βοηθά στη δημιουργία και δοκιμή του λογισμικού σε συνεχή βάση, βοηθώντας τους προγραμματιστές να ενσωματώνουν νέες αλλαγές στο έργο τους και επιτρέποντας στους χρήστες να έχουν πρόσβαση σε νέες εκδόσεις. Επιπλέον, το Jenkins επιτρέπει την αυτόματη παράδοση του λογισμικού, ενσωματώνοντάς το με διάφορες τεχνολογίες δοκιμής και ανάπτυξης (Smart, J. F. 2011).

**GitLab CI/CD:** Ενσωματωμένο μέσα στο GitLab, το GitLab CI/CD παρέχει μια ολοκληρωμένη λύση για τη διαχείριση του κώδικα πηγής, CI και CD. Προσφέρει έλεγχο των εκδόσεων, αναθεώρηση του κώδικα και αυτοματοποίηση της διαδικασίας παράδοσης σε ένα μόνο πλαίσιο (GitLab,2023).

**TeamCity:** Το TeamCity είναι ένα προϊόν της JetBrains που προσφέρει μια ισχυρή πλατφόρμα CI/CD. Υποστηρίζει διάφορες διαμορφώσεις κατασκευής και δοκιμής, προσφέρει ενσωμάτωση με δημοφιλείς συστήματα ελέγχου εκδόσεων και παρέχει εκτενείς δυνατότητες αναφοράς και ανάλυσης (JetBrains 2023).

Αυτά τα εργαλεία CI/CD επιτρέπουν στις ομάδες ανάπτυξης να αυτοματοποιήσουν τις διαδικασίες κατασκευής, δοκιμής και ανάπτυξης, βελτιώνοντας έτσι την ταχύτητα και την αξιοπιστία της παράδοσης λογισμικού ωστόσο χρήζει ιδιαίτερης αναφοράς το Azure DevOps το οποίο διανέμετε από την Microsot.

Το **Azure DevOps**, είναι μια πολυπλευρική πλατφόρμα βασισμένη στο νέφος που διαδραματίζει ένα κεντρικό ρόλο στις σύγχρονες πρακτικές ανάπτυξης λογισμικού. Καλύπτει ένα ευρύ φάσμα υπηρεσιών DevOps για να εξυπηρετήσει τον πλήρη κύκλο ανάπτυξης λογισμικού. Με το Azure DevOps, οι οργανισμοί μπορούν να σχεδιάσουν, αναπτύξουν, δοκιμάσουν, και να παρακολουθήσουν απροβλημάτιστα τα έργα λογισμικού τους. Αυτή η πλατφόρμα είναι σχεδιασμένη για να διευκολύνει τη συνεργασία και τη ροή των εργασιών για τις ομάδες ανάπτυξης και τις λειτουργίες αυτής, ευθυγραμμίζοντάς τις προς κοινούς στόχους και αντικείμενα δράσης. Η δέσμευση της Microsoft για την προώθηση των πρακτικών DevOps είναι εμφανής μέσω του Azure DevOps, που επιτρέπει στους οργανισμούς να παραδίδουν λογισμικό με ταχύτητα, αποτελεσματικότητα και ποιότητα (Microsoft 2023). Ένα από τα βασικά στοιχεία του Azure DevOps είναι το Azure Pipelines. Αυτό το εργαλείο επιτρέπει την υλοποίηση των πρακτικών συνεχούς ενσωμάτωσης (CI) και συνεχούς παράδοσης (CD). Το Azure Pipelines αυτοματοποιεί διάφορες φάσεις της διαδικασίας παράδοσης λογισμικού, συμπεριλαμβανομένης της κατασκευής, της δοκιμής και της ανάπτυξης εφαρμογών. Προσφέρει ευελιξία υποστηρίζοντας μια ευρεία γκάμα γλωσσών προγραμματισμού, πλατφορμών και στόχων ανάπτυξης. Η ενσωμάτωση με υπηρεσίες Azure και εργαλεία τρίτων ενισχύει περαιτέρω τη

δύναμη και την ευελιξία των αγωγών CI/CD, καθιστώντας αξιόπιστη επιλογή για ποικίλα έργα.

Το Azure Boards αποτελεί αναπόσπαστο μέρος του Azure DevOps, προσφέροντας δυνατότητες και ευελιξία στην διαχείριση των έργων. Επιπλέον οι ομάδες μπορούν να σχεδιάζουν, παρακολουθούν και διαχειρίζονται αποτελεσματικά τα αντικείμενα εργασίας τους, όπως για παράδειγμα τα «sprints» και τα «backlog». Τα προσαρμοσμένα ταμπλό Kanban και τα εργαλεία σχεδιασμού sprints διευκολύνουν τις «agile» μεθόδους και τη συνεργατική ομαδική εργασία. Το Azure Boards ως εργαλείο συνίσταται για ομάδες που στοχεύουν στον συντονισμό των προσπαθειών τους και στην αποτελεσματική διαχείριση των προθεσμιών τους.

Από την άλλη πλευρά, το Azure Repos είναι η υπηρεσία ελέγχου εκδόσεων του Azure DevOps που υποστηρίζει τόσο το Git όσο και το Team Foundation Version Control (TFVC). Παρέχει ένα προσβάσιμο και ασφαλές περιβάλλον για τη φιλοξενία και διαχείριση αποθετηρίων του πηγαίου κώδικα. Η ενσωμάτωση με το Azure Pipelines διασφαλίζει την ομαλή μετάβαση των αλλαγών του κώδικα στην CI/CD διαδικασία ενισχύοντας την ανάπτυξη και την παράδοση.

Ως εκ τούτου, σύμφωνα με τις ακόλουθες οδηγίες μπορείτε να δημιουργήσετε και να παραμετροποιήσετε ένα Project στο Azure DevOps ώστε να δημιουργηθεί η διαδικασία CI/CD για την αυτοματοποιημένη ενημέρωση του Camunda Platform σε Kubernetes καθώς και τον υπόλοιπων εφαρμογών που θα εγκατασταθούν σε αυτό.

### **Βήμα 1:** Δημιουργία έργου (Project) Azure DevOps:

Αρχικά το πρώτο βήμα για την δημιουργία μιας CI/CD διαδικασίας είναι η δημιουργία ενός έργου (Project). Το Azure DevOps παρέχει ένα πλήρες σύνολο εργαλείων για τη διαχείριση του κύκλου ζωής ανάπτυξης λογισμικού. Στο έργο, διαμορφώνονται βασικές ρυθμίσεις όπως η επιλογή του αποθετηρίου που θα φιλοξενεί τον κώδικα της εφαρμογής του Camunda. Αυτό το αποθετήριο μπορεί να βρίσκεται στην πλατφόρμα του Azure Repos.

### **Βήμα 2:** Δημιουργία Αγωγού Κατασκευής (Build Pipeline):

Ένα σημαντικό στοιχείο του αγωγού (Pipeline) είναι ο αγωγός κατασκευής (build Pipeline), είναι υπεύθυνος για τον συντονισμό της εφαρμογής του Camunda και τη δημιουργία του ως εικόνα δοχείου (container). Για να δημιουργήσετε αυτόν τον αγωγό, χρησιμοποιήστε τη διαμόρφωση ενός YAML αρχείου, που σας επιτρέπει να ορίσετε ολόκληρη τη διαδικασία δημιουργία της εφαρμογής ως κώδικα. Μέσα σε



αυτόν τον αγωγό, θα καθορίσετε τα απαραίτητα βήματα κατασκευής προσαρμοσμένα στην εφαρμογή Camunda, εξασφαλίζοντας ότι όλες οι εξαρτήσεις επιλύονται και ο κώδικάς σας συντίθεται σωστά. Επιπλέον, συμπεριλάβετε ένα βήμα δημιουργίας Docker για τη συμπύκνωση της εφαρμογής σε εικόνα δοχείου Docker. Είναι σημαντικό να καθορίσετε το αποθετήριο Docker όπου θα αποθηκεύεται αυτή η εικόνα, το οποίο μπορεί να είναι ένα ιδιωτικό αποθετήριο ή ένα αποθετήριο δοχείων βασισμένο στον νέφος, όπως το Azure Container Registry.

### **Βήμα 3:** Διαμόρφωση Αγωγού Κυκλοφορίας (Release Pipeline):

Αφού η εφαρμογή Camunda είναι σε εικόνα δοχείου, ο αγωγός κυκλοφορίας αναλαμβάνει την μεταφορά των παραγόμενων αρχείων όπως αυτά δημιουργήθηκαν κατά την κατασκευή της εικόνας στο Kubernetes cluster. Σε αυτόν τον αγωγό, θα διαμορφωθούν διάφορα στάδια όπως ενεργοποιητές και αντικείμενα. Για να εγκαταστήσετε την εφαρμογή σας στο Kubernetes, θα πρέπει να χρησιμοποιήσετε μια εργασία αναπτύξεων (Deployment Manifest) που περιλαμβάνει μια δήλωση για το Kubernetes. Συνδέστε αυτήν την εργασία στην κατάλληλη σύνδεση (Service Connection) για το Kubernetes, η οποία θα περιέχει τα απαραίτητα διαπιστευτήρια και πληροφορίες πρόσβασης για το συστηματικό περιβάλλον του Kubernetes. Επίσης, ορίστε το χώρο (namespace) του Kubernetes όπου θα γίνει η εγκατάσταση της εφαρμογής Camunda.

### **Βήμα 4:** Ενεργοποιητές και Μεταβλητές Αγωγού (Pipeline Triggers and Variables):

Για να διαχειριστείτε αποτελεσματικά τη διαδικασία της συνεχή ενσωμάτωσης και παράδοσης, πρέπει να προσαρμόσετε τους "ενεργοποιητές" και τις "μεταβλητές". Οι ενεργοποιητές καθορίζουν πότε θα ξεκινήσει αυτόματα η διαδικασία ενσωμάτωσής. Για παράδειγμα, μπορείτε να ρυθμίσετε ενεργοποιητές για να ξεκινήσετε έναν αυτόματο κύκλο κατασκευής του κώδικα κάθε φορά που γίνεται καταχώρηση κώδικα στο αποθετήριο ή όταν δημιουργείται μια νέα έκδοση. Αυτοί οι ενεργοποιητές διασφαλίζουν ότι οι διαδικασίες είναι πάντα συγχρονισμένες με τις αναπτυξιακές δραστηριότητες σας. Επιπλέον, πρέπει να ορίσετε μεταβλητές και ρυθμίσεις περιβάλλοντος που εξαρτώνται από τις διαδικασίες σας. Αυτές οι μεταβλητές μπορεί να περιλαμβάνουν ευαίσθητες πληροφορίες, κλειδιά API ή περιβαλλοντικές ρυθμίσεις καθώς διαπιστευτήρια. Η σωστή διαμόρφωση αυτών των στοιχείων συμβάλλει στη διασφάλιση μιας ομαλής και ασφαλούς εκτέλεσης των διαδικασιών.

### **Βήμα 5:** Εκτέλεση Pipeline:

Αφού έχουν διαμορφωθεί οι διαδικασίες , μπορείτε να εκτελέσετε τον αγωγό (Pipeline). Ένας "αγωγός κατασκευής" ενεργοποιεί τη δημιουργία της εφαρμογής Camunda και τη δημιουργία ενός εικονικού δοχείου (container) για αυτήν. Έπειτα, αυτή η εικόνα αποθηκεύεται στον καθορισμένο χώρο αποθήκευσης Docker (image registry). Μετά από μια επιτυχημένη κατασκευή, μπορείτε να ενεργοποιήσετε έναν "αγωγό απελευθέρωσης" (release pipeline) για να μεταφέρετε την εφαρμογή Camunda στον χώρο της υποδομής του Kubernetes. Για να διασφαλίσετε μια ομαλή διαδικασία ανάπτυξης, είναι σημαντικό να έχετε τις απαραίτητες άδειες και πρόσβαση στον Kubernetes cluster. Βεβαιωθείτε ότι οι ρυθμίσεις σύνδεσης με την υπηρεσία Kubernetes και τον χώρο ονομάτων (Namespaces) είναι σωστά ρυθμισμένες στο Azure DevOps. Ακολουθώντας αυτά τα βήματα, δημιουργείτε έναν αποτελεσματικό αγωγό για τον συνεχή ενσωμάτωση και παράδοση που επιτρέπει την ταχεία ανάπτυξη, κατασκευή και δημιουργία της εφαρμογής του Camunda στο περιβάλλον του Kubernetes. Η προσαρμογή κάθε μέρους της διαδικασίας στις ανάγκες του έργου είναι ζωτικής σημασίας για την επίτευξη μιας αποτελεσματικής ροής εργασιών.

## Βιβλιογραφία

- Aalst, W. M. P. (2016). *Process mining: Data science in action*. Springer.
- Allweyer, T. (2016). *BPMN 2.0: introduction to the standard for business process modeling*. BoD—Books on Demand.
- Appian. (2023). Appian BPM. Ανακτήθηκε από <https://www.appian.com/platform/bpm/>
- Bala, A., Ghosh, S., & Nagwani, N. K. (2019). Business Process Management Systems: Advantages, Limitations, and Deployment Strategies. In *Handbook of Research on Information Architecture and Management in Modern Organizations* (pp. 80-101). IGI Global.
- BPMN.org. (2021). Προδιαγραφές BPMN. <https://www.bpmn.org/specifications/>
- Camunda (2023). Camunda BPM - Workflow Automation and BPMN Engine. Ανακτήθηκε από <https://camunda.com/>
- Chen, L., Xiong, J., & Yu, W. (2019). A blockchain-based approach for traceability and transparency in the supply chain. In *2019 IEEE 39th International Conference on Distributed Computing Systems (ICDCS)* (pp. 206-216). IEEE.
- Chinosi, M., & Trombetta, A. (2012). BPMN: An introduction to the standard. *Computer Standards & Interfaces*, 34(1), 124-134.
- Damij, N., Damij, T., Grad, J., & Jelenc, F. (2008). A methodology for business process improvement and IS development. *Information and software technology*, 50(11), 1127-1141.
- Das, M., Deb, M., Wilkins, M., 2011. *Oracle Business Process Management Suite 11g Handbook*. McGraw-Hill Osborne Media.
- Davenport, T. H. (1993). *Process innovation: Reengineering work through information technology*. Harvard Business Press.
- Davenport, T. H. (2006). *Competing on Analytics*. Harvard Business Review.
- Davenport, T. H., & Harris, J. (2007). *Competing on Analytics: The New Science of Winning*. Harvard Business Review Press.
- Dijkman, R., Dumas, M., & Ouyang, C. (2011). Semantics and Analysis of Business Process Models in BPMN. *Information and Software Technology*, 53(5), 535-550.
- Dumas, M., La Rosa, M., Mendling, J., & Reijers, H. A. (2013). *Fundamentals of business process management*. Springer.
- Duvall, P., Matyas, S., & Glover, A. (2007). "Continuous Integration: Improving Software Quality and Reducing Risk." Addison-Wesley Professional.

GitLab (2023). GitLab CI/CD. Ανακτήθηκε από <https://about.gitlab.com/stages-devops-lifecycle/continuous-integration/>

Hammer, M. (2007). The Process Audit. Harvard Business Review.

Hasselbring, W., & Shakeri, H. (2019). Monitoring Business Process Compliance: Challenges, Solutions, and Tools. In Enterprise, Business-Process and Information Systems Modeling (pp. 83-98). Springer.

He, W., Zhang, J., Meng, F., & Li, Q. (2020). Integrating business process management and blockchain for supply chain traceability: A systematic review and research agenda. International Journal of Information Management, 50, 302-315.

Heuvel, W. J., & Jansen-Vullers, M. H. (2017). Towards efficient orchestration of scientific workflows in web portals. Future Generation Computer Systems, 68, 334-348.

Humble, J., & Farley, D. (2010). "Continuous Delivery: Reliable Software Releases through Build, Test, and Deployment Automation." Addison-Wesley Professional.

IBM. (2023). IBM Business Process Manager. Ανακτήθηκε από <https://www.ibm.com/products/business-process-manager>

Jeston, J., & Nelis, J. (2014). Business process management: Practical guidelines to successful implementations. Routledge.

JetBrains (2023). TeamCity. Retrieved from <https://www.jetbrains.com/teamcity/>

Kumari, A., Padi, N. R., & Chauhan, M. (2020). An improved architecture for business process management systems using RESTful services. Information Systems Frontiers, 22(4), 947-967.

Kumari, A., Sareen, S., & Kumar, M. (2020). Design and Analysis of a Business Process Automation Solution: A Case Study. International Journal of Applied Engineering Research, 15(9), 1715-1720.

Lam, V. S. (2009). Equivalences of BPMN processes. Service Oriented Computing and Applications, 3, 189-204.

Leoni, G., Vincenzi, A., & Riccucci, C. (2020). Open-Source Process Engines: A Systematic Mapping Study. Information and Software Technology, 120, 106263.

Leopold, H., Mendling, J., & Günther, O. (2015). Learning from quality issues of BPMN models from industry. IEEE software, 33(4), 26-33.

McAfee, A., Brynjolfsson, E. (2017). Machine, Platform, Crowd: Harnessing Our Digital Future. W. W. Norton & Company.

Mendling, J., Weber, I., & van der Aalst, W. M. (2008). Visualizing process compliance in terms of deviations and performance. International Journal of Business Process Integration and Management, 3(2), 124-133.

Microsoft (2023) Azure Devops. Ανακτήθηκε από <https://azure.microsoft.com/en-us/products/devops>

Nasiri, F., Pishvae, M. S., & Rad, S. (2019). Designing a sustainable closed-loop supply chain network based on Triple Bottom Line approach: A comparison of metaheuristic hybridization techniques. *Journal of Cleaner Production*, 238, 117828.

Nasiri, M. M., Pishvae, M. S., & Rad, S. (2019). Business process management in the service industry: Challenges, solutions, and future directions. *Computers & Industrial Engineering*, 131, 1-17.

OMG, 2011. Business process model and notation 2.0. Tech. rep., Object Management Group, Washington DC, USA

OMG. (2017). Decision Model and Notation (DMN) Version 1.2. <https://www.omg.org/spec/DMN/1.2/>

Oracle. (2023). Oracle Business Process Management Suite. Ανακτήθηκε από <https://www.oracle.com/middleware/technologies/bpm.html>

Owen, M., & Raj, J. (2005). BPMN and business process management; Introduction to the new business process modeling Standard (2003). *BPMN and Business Process Management-Introduction to the New Business Process Modeling Standard 2003*, 13.

Pegasystems. (2023). Pega Platform. Ανακτήθηκε από <https://www.pega.com/products/pega-platform>

Peters, M. (2018). *Business Process Management: Concepts, Languages, Architectures*. Springer.

Peters, M. A. (2018). Retailer-driven sustainable supply chain management: Walmart's pioneering strategies and methods in the USA. *Sustainability*, 10(9), 3049.

Pritchard, D., Juttner, U., & Johnston, R. B. (2016). A framework for business process management. *Journal of Computer Information Systems*, 56(2), 101-109.

Rademakers, T., 2012. *Activiti in Action: Executable business processes in BPMN 2.0*. Manning Publications Co

Rao, S. S., & Metts, G. (2007). Implementing and managing business process reengineering. *Total Quality Management & Business Excellence*, 18(4), 417-430.

Recker, J., Rosemann, M., & Green, P. (2015). Business process modeling: A comparative analysis. *Journal of the Association for Information Systems*, 16(2), 129-170.

Richardson, C., Cullen, A., McGovern, S., & Lynch, D. (2015). *The Forrester Wave™: BPM Platforms For Digital Business, Q4 2015*. Forrester.

- Ross, J. W., Beath, C., Quaadgras, A. (2013). "Digital Innovation Leadership: A Research Agenda." *Journal of Management Information Systems*, 30(1), 123-139.
- Sarkis, J. (2020). Business process management and the circular economy: A review and agenda for research. *Resources, Conservation and Recycling*, 161, 104962.
- Schroeder, R. G., Linderman, K., Liedtke, C., & Choo, A. S. (2011). Six sigma: Definition and underlying theory. *Journal of Operations Management*, 29(5), 435-452.
- Silver, E. A., Pyke, D. F., & Peterson, R. (2009). *Inventory management and production planning and scheduling*. CRC Press.
- Smart, J. F. (2011). *Jenkins: The Definitive Guide: Continuous Integration for the Masses*. " O'Reilly Media, Inc."
- Smith, J. (2020). *Digital Transformation: A Roadmap for Billion-Dollar Organizations*. Wiley.
- van der Aalst, W. M. P. (2013). *Business process management: A comprehensive survey*. ISRN Software Engineering, 2013.
- van der Aalst, W. M., Reijers, H. A., & Mendling, J. (2019). *Data Science in Action: Redesigning Business Processes Using Process Mining*. MIT Press.
- von Rosing, M., White, S., Cummins, F., & de Man, H. (2015). *Business Process Model and Notation-BPMN*
- Westerman, G., Bonnet, D., McAfee, A. (2014). *Leading Digital: Turning Technology into Business Transformation*. Harvard Business Review Press.
- Wheelwright, S. C., & Clark, K. B. (1992). *Revolutionizing Product Development: Quantum Leaps in Speed, Efficiency, and Quality*. Free Press.
- White, S. A. (2004). *Introduction to BPMN; BPTrends*.
- Wynn, M. T., & Eckman, B. A. (2017). Business Process Management Suites (BPMS) and their effect on service innovation capabilities. *International Journal of Information Management*, 37(2), 53-65.
- Wynn, M., & Eckman, P. (2017). *Evaluating the Total Cost of Ownership of Camunda's BPM Platform*. Camunda White Paper.
- Yan, J., Gorton, I., & Zhang, H. (2018). A Comprehensive Evaluation of Open Source BPM Platforms. *Journal of Systems and Software*, 142, 176-196.
- Yousfi, A., Bauer, C., Saidi, R., & Dey, A. K. (2016). uBPMN: A BPMN extension for modeling ubiquitous business processes. *Information and Software Technology*, 74, 55-68.

Zur Muehlen, M., & Indulska, M. (2010). Towards a conceptual framework for business process flexibility. In *Business Process Management Workshops* (pp. 235-247). Springer.

## Πηγές Εικόνων

- Εικόνα 1: Overview of Verification Tools for Business Process Models, 2017 Federated Conference on Computer Science and Information Systems. Anna Suchenia, Piotr Wiśniewski, Antoni Ligęza
- Εικόνα 2: Towards a Methodology for Business Process Revision under Norm and Outcome Compliance, 2014 Simone Scannapieco
- Εικόνα 3: Designing a Framework of Components to Support Patient Engagement in Research, 2019 2019
- Εικόνα 4: An Artifact-Driven Approach to Monitor Business Processes Through Real-World Objects, 2017, Giovanni Meroni Claudio Di Ciccio Jan Men