

**ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΙΡΑΙΩΣ**

**ΤΜΗΜΑ ΒΙΟΜΗΧΑΝΙΚΗΣ ΔΙΟΙΚΗΣΗΣ & ΤΕΧΝΟΛΟΓΙΑΣ**



**ΠΡΟΓΡΑΜΜΑ ΜΕΤΑΠΤΥΧΙΑΚΩΝ ΣΠΟΥΔΩΝ ΣΤΗ ΒΙΟΜΗΧΑΝΙΚΗ ΔΙΟΙΚΗΣΗ ΚΑΙ ΤΕΧΝΟΛΟΓΙΑ**

---

**ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ**

**ΤΙΤΛΟΣ**

**«ΑΝΑΛΥΣΗ ΔΕΔΟΜΕΝΩΝ ΓΙΑ ΕΥΕΛΙΚΤΗ ΔΙΑΧΕΙΡΙΣΗ ΕΡΓΩΝ  
ΠΛΗΡΟΦΟΡΙΚΗΣ»**

**ΟΝΟΜΑΤΕΠΩΝΥΜΟ: ΚΟΥΤΣΟΠΟΥΛΟΣ ΑΘΑΝΑΣΙΟΣ  
ΕΠΙΒΛΕΠΩΝ ΚΑΘΗΓΗΤΗΣ: Δρ. ΕΙΡΗΝΑΚΗΣ ΠΑΥΛΟΣ**

**Πειραιάς 2021**

Πανεπιστήμιο Πειραιώς, Τμήμα Βιομηχανικής Διοίκησης & Τεχνολογίας

Αθανάσιος Κουτσόπουλος

© 2021 – Με την επιφύλαξη παντός δικαιώματος

Η εργασία αυτή είναι πρωτότυπη και εκπονήθηκε αποκλειστικά και μόνο για την απόκτηση του συγκεκριμένου μεταπτυχιακού τίτλου.

Τα πνευματικά δικαιώματα χρησιμοποίησης του μη πρωτότυπου υλικού ΜΔΕ ανήκουν στο μεταπτυχιακό φοιτητή και το επιβλέπον μέλος ΔΕΠ εις ολόκληρο, δηλαδή εκάτερος μπορεί να κάνει χρήση αυτών χωρίς τη συναίνεση άλλου. Τα πνευματικά δικαιώματα χρησιμοποίησης του πρωτότυπου μέρους ΜΔΕ ανήκουν στον μεταπτυχιακό φοιτητή και τον επιβλέποντα από κοινού, δηλαδή δεν μπορεί ο ένας από τους δύο να κάνει χρήση αυτού χωρίς τη συναίνεση του άλλου. Κατ' εξαίρεση, επιτρέπεται η δημοσίευση του πρωτότυπου μέρους της διπλωματικής εργασίας σε επιστημονικό περιοδικό ή πρακτικά συνεδρίου από τον ένα εκ των δύο, με την προϋπόθεση ότι αναφέρονται τα ονόματα και των δύο (ή των τριών σε περίπτωση συνεπιβλέποντα) ως συν-συγγραφέων. Στην περίπτωση αυτή προηγείται γραπτή ενημέρωση του μη συμμετέχοντα στη συγγραφή του επιστημονικού άρθρου. Δεν επιτρέπεται η κατά οποιοδήποτε τρόπο δημοσιοποίηση υλικού το οποίο έχει δηλωθεί εγγράφως ως απόρρητο.

Η παρούσα διπλωματική εργασία αφιερώνεται  
στον αδερφό μου Γιάννη.

## Περίληψη

Η εταιρεία κατασκευής λογισμικού εμπορικής και λογιστικής διαχείρισης Softone Technologies, στα πλαίσια της ανάπτυξής της, αποφάσισε να εφαρμόσει την μεθοδολογία agile. Σκοπός της υιοθέτησης της μεθοδολογίας αυτής, ήταν ο καλύτερος προγραμματισμός των εργασιών ανάπτυξης και συντήρησης του βασικού προϊόντος της, του Softone ERP καθώς και η βελτίωση της επικοινωνίας μεταξύ των διαφόρων αρμοδίων τμημάτων με το τμήμα έρευνας και ανάπτυξης που είναι ως επί το πλείστον το υπεύθυνο τμήμα παραγωγής του προϊόντος. Η παρούσα εργασία μελετά την επίδραση της εφαρμογής της μεθοδολογίας agile στην απόδοση του τμήματος έρευνας και ανάπτυξης (R&D).

Για τις ανάγκες της συγκεκριμένης μελέτης ορίστηκαν δείκτες απόδοσης (KPIs) βασισμένοι σε πραγματικά δεδομένα από τις πλατφόρμες διαχείρισης των αιτημάτων. Από την πρώτη πλατφόρμα, που αποτελεί το κατεξοχήν σύστημα καταγραφής αιτημάτων (ticketing system) του εξωτερικού περιβάλλοντος (συνεργάτες και πελάτες), αντλήθηκαν δεδομένα για τα τελευταία έξι χρόνια λειτουργίας της εταιρείας (2015-2020). Από την δεύτερη πλατφόρμα, που αποτελεί το σύστημα καταγραφής, οργάνωσης και παρακολούθησης των αιτημάτων σε όρους agile, αντλήθηκαν δεδομένα για τα τρία τελευταία χρόνια (2018-2020), στα οποία εφαρμόζεται ήδη η μέθοδος στο τμήμα έρευνας και ανάπτυξης.

Η μελέτη όλων των παραπάνω στοιχείων και τα αποτελέσματα που ανέκυψαν από τους δείκτες απόδοσης, έδειξαν ότι η εφαρμογή της μεθοδολογίας agile είχε πολλαπλά οφέλη για το τμήμα έρευνας και ανάπτυξης και κατά συνέπεια για την συνολική εικόνα της εταιρείας. Ορισμένα από τα οφέλη που παρουσιάζονται είναι: η αύξηση του αριθμού των υλοποιημένων νέων δυνατοτήτων από τα μέλη κάθε scrum ομάδας ανά έτος, η καλύτερη ενσωμάτωση των νέων προγραμματιστών που οδήγησε με τη σειρά της στην ταχύτερη αύξηση των δυνατοτήτων του τμήματος, καθώς και η σημαντική μείωση της δημιουργίας σφαλμάτων που φτάνουν στον τελικό πελάτη.

## Abstract

As part of its development, the software company ‘Softone Technologies’, decided to apply agile methodology. The purpose of adopting this methodology was the better planning of the development and maintenance of its main product, the Softone ERP, as well as the improvement of the communication between the various departments and the Research and Development department which is mostly responsible for the productive part of the company. This thesis examines the effect of the application of the agile methodology to the efficiency of the Research and Development department.

For the needs of this study, key performance indicators (KPIs) were developed based on real data from the request management platforms. From the first platform, which is the foremost ticketing system of external environment (partners and customers), data were extracted for the period 2015-2020 - the last six operational years of the company. From the second platform, which is the system of recording, organizing and monitoring requests in agile terms, data were extracted for the period 2018-2020 - the last three years, that this method has been applied to the Research and Development department.

The study of all the above and the results obtained from the key performance indicators, showed that the application of the agile methodology had multiple benefits for the Research and Development department and consequently for the overall image of the company. Some of the benefits presented are; the increase in the number of the newly implemented specs by the developer of each scrum team per year, the better integration of the newly acquired developers which in its turn lead to a faster increase in the potential of the Research and Development department, as well as the significant reduction of errors reaching the final customer.

## Ευχαριστίες

Η παρούσα διπλωματική εργασία είναι αποτέλεσμα προσπάθειας μίας ολόκληρης χρονιάς και θα ήταν αδύνατο να εκπονηθεί χωρίς την βοήθεια κάποιων ανθρώπων τους οποίους και θα ήθελα να ευχαριστήσω.

Αρχικά, τον Επιβλέποντα Καθηγητή **κ. Παύλο Ειρηνάκη**, επίκουρο καθηγητή του τμήματος Βιομηχανικής Διοίκησης & Τεχνολογίας του Πανεπιστημίου Πειραιώς, ο οποίος με τίμησε με την εμπιστοσύνη του και μου έδωσε τη δυνατότητα να ασχοληθώ με ένα τόσο ενδιαφέρον θέμα.

Ιδιαίτερα ευχαριστώ τον **κ. Άγγελο Πεδιώτη**, Product Manager στην εταιρεία Softone Technologies για την καθοδήγησή του και τις πολύτιμες συμβουλές του κατά την διάρκεια της ενασχόλησής μου με την εργασία.

Επιπλέον, ευχαριστώ θερμά τον προϊστάμενο μου, **κ. Γιώργο Τσομόκο**, διευθυντή R&D Core της εταιρείας Softone Technologies καθώς και τον γενικό διευθυντή της εταιρείας **κ. Αντώνη Καραντώνη** γιατί χωρίς την στήριξη της εταιρείας δεν θα είχα πραγματοποιήσει τις σπουδές αυτές.

Τέλος, θα ήθελα να ευχαριστήσω την οικογένειά μου για την υποστήριξη που μου παρείχαν και που εξακολουθούν να μου παρέχουν σε όλη τη διάρκεια των φοιτητικών μου χρόνων και για τις πολλές θυσίες που έκαναν προκειμένου να φτάσω έως εδώ καθώς και την μέλλουσα σύζυγό μου Ξανθίπη που χρειάστηκε να κάνει πολλή υπομονή το διάστημα των σπουδών μου.

## Περιεχόμενα

Κεφάλαιο 1°	1
1.1 Εισαγωγή	1
1.2 Δομή Εργασίας	2
Κεφάλαιο 2°	4
2.1 Η μέθοδος Agile	4
2.2 Agile Manifesto	5
2.2.1 Οι αξίες.....	6
2.2.2 Οι βασικές αρχές.....	7
2.3 Pros-Cons	8
2.3.1 Τα Θετικά .....	9
2.3.2 Τα Αρνητικά .....	10
2.4 Οι πιο διαδεδομένες Agile πρακτικές- μέθοδοι	11
2.4.1 Adaptive software development (ASD) .....	11
2.4.2 Extreme Programming (XP).....	12
2.4.3 Crystal Clear methods .....	13
2.4.4 Feature-driven development (FDD) .....	14
2.4.5 Dynamic systems development method (DSDM) .....	14
2.4.6 Lean Software Development (LSD) .....	15
2.4.7 Kanban .....	16
2.4.8 Scrum .....	17
2.4.8.1 Βασικές Οντότητες	20
2.4.8.2 Πρακτικές	22
Κεφάλαιο 3°	25
3.1 Μελέτη περίπτωσης: Softone Technologies SA	25
3.2 Εταιρική δομή	26
3.3 Agile προσέγγιση	28
3.3.1 Εισαγωγή.....	28
3.3.2 Οντότητες.....	29
3.3.3 Διαδικασίες.....	30
Κεφάλαιο 4°	32
4.1 Συλλογή Δεδομένων Εξωτερικού Περιβάλλοντος	32
4.2 Ποσοτικοί Δείκτες απόδοσης (KPIs)	34



4.2.1 Αριθμός νέων προδιαγραφών ανά εγκατάσταση .....	34
4.2.1.1 Ανάγκη	34
4.2.1.2 Ανάλυση	35
4.2.1.3 Συμπεράσματα	35
4.2.2 Αριθμός υλοποιημένων προδιαγραφών ανά εγκατάσταση .....	36
4.2.2.1 Ανάγκη	36
4.2.2.2 Ανάλυση	36
4.2.2.3 Συμπεράσματα	37
4.2.3 Αριθμός καταγεγραμμένων δυσλειτουργιών ανά εγκατάσταση .....	37
4.2.3.1 Ανάγκη	37
4.2.3.2 Ανάλυση	37
4.2.3.3 Συμπεράσματα	38
4.2.4 Δυσλειτουργίες σε σχέση με το μέγεθος του κώδικα .....	38
4.2.4.1 Ανάγκη	38
4.2.4.2 Ανάλυση	39
4.2.4.3 Συμπέρασμα	39
4.2.5 Αριθμός επιλυμένων δυσλειτουργιών ανά εγκατάσταση.....	40
4.2.5.1 Ανάγκη	40
4.2.5.2 Ανάλυση	40
4.2.5.3 Συμπεράσματα	41
4.2.6 Ανάπτυξη - Αποσφαλμάτωση ανά προγραμματιστή.....	41
4.2.6.1 Ανάγκη	41
4.2.6.2 Ανάλυση	42
4.2.6.3 Συμπεράσματα	43
Κεφάλαιο 5 <sup>ο</sup>	44
5.1 Συλλογή Δεδομένων Εσωτερικού Περιβάλλοντος	44
5.2 Ποσοτικοί δείκτες απόδοσης(KPIs)	46
5.2.1 Ποσοστό user stories που υλοποιήθηκαν .....	46
5.2.1.1 Ανάγκη	46
5.2.1.2 Ανάλυση	46
5.2.1.3 Συμπεράσματα	47
5.2.2 Ποσοστό δυσλειτουργιών που διορθώθηκαν .....	47
5.2.2.1 Ανάγκη	47

5.2.2.2 Ανάλυση	47
5.2.2.3 Συμπεράσματα	48
5.2.3 Users Stories – Σφάλματα ανά Scrum Team.....	48
5.2.3.1 Ανάγκη	48
5.2.3.2 Ανάλυση	49
5.2.3.3 Συμπεράσματα	49
5.2.4 Απόδοση των Scrum Ομάδων ανά sprint .....	50
5.2.4.1 Ανάγκη	50
5.2.4.2 Ανάλυση	50
5.2.4.3 Συμπεράσματα	51
Κεφάλαιο 6°	52
6.1 Συμπεράσματα	52
6.2 Μελλοντική μελέτη - προτάσεις βελτίωσης	54
Βιβλιογραφία	56

## Κατάλογος Εικόνων

Εικόνα 1: Agile Φιλοσοφία	4
Εικόνα 2: Agile Manifesto	5
Εικόνα 3: Agile Μεθοδολογίες	11
Εικόνα 4: Adaptive software development	12
Εικόνα 5: Extreme Programming (XP)	13
Εικόνα 6: Feature-driven development (FDD)	14
Εικόνα 7:Lean Software Development (LSD)	15
Εικόνα 8: Kanban Board example	16
Εικόνα 9:Scrum Process	18
Εικόνα 10: Softone ticketing system	32
Εικόνα 11: Kanban Board, Youtrack	44

## Κατάλογος Διαγραμμάτων

Διάγραμμα 1: Ποσοστιαία μεταβολή ενεργών εγκαταστάσεων	33
Διάγραμμα 2: Αιτήσεις για προσθήκη νέων προδιαγραφών	35
Διάγραμμα 3: Υλοποίηση νέων προδιαγραφών	36
Διάγραμμα 4: Δυσλειτουργίες ανά 100 εγκαταστάσεις	38
Διάγραμμα 5: Δυσλειτουργίες ανά 1000 γραμμές κώδικα	39
Διάγραμμα 6: Επιδιορθωμένες δυσλειτουργίες ανά 100 εγκαταστάσεις	41
Διάγραμμα 7: Νέες προδιαγραφές - Διορθώσεις σφαλμάτων/ προγραμματιστή	42
Διάγραμμα 8: Ποσοστό υλοποιημένων user stories	46
Διάγραμμα 9: Ποσοστό διορθωμένων σφαλμάτων	48
Διάγραμμα 10: Μέσος όρος Ολοκληρωμένων users stories - σφαλμάτων ανά ομάδα	49
Διάγραμμα 11: Μέσος όρος velocity ανά sprint	51

## Κεφάλαιο 1<sup>ο</sup>

### 1.1 Εισαγωγή

Στη σημερινή εποχή οι επιχειρήσεις καλούνται να προσαρμοστούν στις ανταγωνιστικές συνθήκες που επικρατούν στην αγορά προκειμένου να βρουν τρόπους εξοικονόμησης κόστους και βελτιστοποίησης των έμψυχων πόρων τους. Ανώτερος στόχος είναι να διατηρήσουν το ανταγωνιστικό τους πλεονέκτημα προσφέροντας προϊόντα και υπηρεσίες σε όσο το δυνατόν καλύτερη τιμή διατηρώντας την ποιότητα σε υψηλά επίπεδα.

Για το λόγο αυτό πολλές εταιρείες - ειδικά στο χώρο της τεχνολογίας που οι αλλαγές γίνονται με ταχύτατους ρυθμούς - καλούνται να βελτιστοποιήσουν τις διαδικασίες τους ώστε να εναρμονίζονται με τις σημερινές ανάγκες. Οι διαδικασίες αυτές αφορούν τόσο την ανάπτυξη και συντήρηση των προϊόντων - υπηρεσιών τους όσο και την συνολική επικοινωνία μεταξύ του εσωτερικού και του εξωτερικού περιβάλλοντος της επιχείρησης.

Η διοίκηση και διαχείριση έργων (project management) αναπτύχθηκε σαν ξεχωριστό γνωστικό πεδίο από την εφαρμογή των αρχών της διοίκησης αλλά και της επιχειρησιακής έρευνας σε διάφορους τομείς εφαρμογής, όπως αυτός των κατασκευών, της μηχανολογίας, της πληροφορικής και των μεγάλων δημόσιων ή και στρατιωτικών προγραμμάτων.

Η ευέλικτη διαχείριση έργων αποτελεί μια δημοφιλή μέθοδο που τα τελευταία χρόνια υιοθετείται από πολλές εταιρείες. Βασίζεται σε αρχές και περιλαμβάνει τεχνικές που βρίσκουν εφαρμογή και σε επιχειρηματικές δραστηριότητες που δεν σχετίζονται απαραίτητα με την ανάπτυξη έργων.

Στόχος της συγκεκριμένης διπλωματικής εργασίας είναι η μελέτη της απόδοσης της εφαρμογής της agile μεθόδου στο τμήμα Έρευνας & Ανάπτυξης, μίας από τις μεγαλύτερες εταιρείες κατασκευής ολοκληρωμένου επιχειρηματικού πληροφοριακού συστήματος της ελληνικής αγοράς, την Softone Technologies. Με την αξιοποίηση δεδομένων από τα βασικά συστήματα διαχείρισης αιτημάτων ανάπτυξης και διόρθωσης (ticketing system) αναπτύχθηκαν και παρουσιάζονται αντιπροσωπευτικοί δείκτες απόδοσης.

Σκοπός της συλλογής και ανάλυσης των δεδομένων του εξωτερικού συστήματος είναι η δημιουργία συμπερασμάτων σχετικά με το πόσο συνέβαλε η εφαρμογή της agile μεθόδου στην

αποτελεσματικότητα του τμήματος στη πάροδο του χρόνου. Αντίστοιχα, μέσα από την ανάλυση των δεδομένων της εσωτερικής πλατφόρμας διαχείρισης αιτημάτων στα πλαίσια της agile μεθόδου προκύπτουν πιο λεπτομερή συμπεράσματα. Τα συμπεράσματα αυτά σχετίζονται με την λειτουργία του τμήματος καθ' όλη τη διάρκεια εφαρμογής της μεθόδου, με τη πιθανή δυναμική του και με τη συμπεριφορά του στο χρόνο, όσο ωριμάζει η φιλοσοφία της ευέλικτης διαχείρισης. Σύμφωνα με τα συμπεράσματα της παρούσας μελέτης μπορεί να γίνει εκτίμηση για το αν άξιζε η αλλαγή τρόπου λειτουργίας με σκοπό την εφαρμογή της συγκεκριμένης μεθοδολογίας και στα υπόλοιπα παραγωγικά τμήματα της εταιρείας.

## 1.2 Δομή Εργασίας

Στη παρούσα διπλωματική εργασία παρουσιάζεται η ευέλικτη διαχείριση έργων (Agile) και η εφαρμογή της στην εταιρεία Softone Technologies, μέσα από έξι κεφάλαια.

Πιο συγκεκριμένα, στο πρώτο κεφάλαιο, γίνεται μια εισαγωγή στην διαχείριση έργων γενικότερα καθώς και αναφορά στην ανάγκη της συγκεκριμένης εταιρείας που οδήγησε στην υιοθέτηση του ευέλικτου μοντέλου διαχείρισης έργων.

Στο δεύτερο κεφάλαιο γίνεται αναλυτική αναφορά στην ευέλικτη διαχείριση έργων. Παρουσιάζεται η ιστορική διαδρομή κατά τη θεμελίωσή της ενώ αναλύονται οι πιο συνηθισμένες μεθοδολογίες με τις οποίες εφαρμόζεται σήμερα.

Στο τρίτο κεφάλαιο, παρουσιάζεται η εταιρική δομή της Softone Technologies SA. Παρουσιάζονται τα βασικότερα τμήματα της εταιρείας καθώς και ο τρόπος που αλληλοεπιδρούν μεταξύ τους. Τέλος, αναλύονται οι οντότητες και οι διαδικασίες της agile μεθοδολογίας τις οποίες έχει υιοθετήσει η εταιρεία.

Στο τέταρτο κεφάλαιο γίνεται η παρουσίαση βασικών ποσοτικών και ποιοτικών δεικτών απόδοσης (KPIs) που σχετίζονται με την ανάπτυξη και συντήρηση του βασικού προϊόντος της εταιρείας. Οι δείκτες αυτοί μελετώνται από την εξωτερική οπτική του τμήματος. Τα δεδομένα τα οποία αναλύονται αφορούν χρονικό διάστημα πριν και μετά την υιοθέτηση της μεθοδολογίας. Αυτό έχει σαν αποτέλεσμα την εξαγωγή ασφαλών συμπερασμάτων σχετικά με την διαφοροποίηση που προέκυψε από την εφαρμογή της μεθόδου.

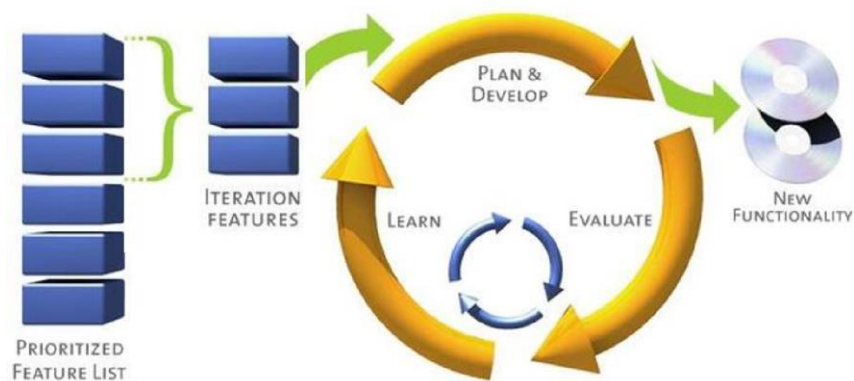
Στο πέμπτο κεφάλαιο παρουσιάζονται και σχολιάζονται βασικοί ποσοτικοί δείκτες απόδοσης (KPIs) που σχετίζονται με την εσωτερική οργάνωση και διαχείριση του τμήματος όσον αφορά τα διάφορα αιτήματα που εισέρχονται από το εξωτερικό περιβάλλον. Τα δεδομένα αφορούν τη χρονική περίοδο που εφαρμόζεται η μεθοδολογία agile στο τμήμα. Αποτέλεσμα της μελέτης είναι η εξαγωγή συμπερασμάτων που αφορούν τη πιθανή εξέλιξη του τμήματος στο πέρασμα του χρόνου, την παρούσα δυναμική του αλλά και ενδεχόμενες βελτιώσεις για ακόμα καλύτερα αποτελέσματα.

Στο έκτο και τελευταίο κεφάλαιο, γίνεται μια σύνοψη των συμπερασμάτων που προέκυψαν από την ανάλυση των δεδομένων και την δημιουργία των δεικτών απόδοσης των δύο προηγούμενων κεφαλαίων. Επιπλέον γίνεται αναφορά σε βελτιώσεις που θα μπορούσαν να επιτευχθούν στην καταγραφή των δεδομένων προκειμένου μελλοντικά να προκύψουν περισσότεροι δείκτες με περισσότερες λεπτομέρειες.

## Κεφάλαιο 2<sup>ο</sup>

### 2.1 Η μέθοδος Agile

Το Agile είναι μια ευέλικτη προσέγγιση, βασισμένη σε επαναλήψεις, για την σταδιακή ανάπτυξη και παράδοση τμημάτων λογισμικού. Από την αρχή του έργου η παράδοση γίνεται σταδιακά σε τμήματα και όχι συγκεντρωτικά στο τέλος. Χωρίζει το έργο σε λειτουργικά κομμάτια (*user stories*), δίνοντάς τους προτεραιότητες και στη συνέχεια τα παραδίδει σε σύντομους κύκλους προκαθορισμένου χρόνου που ονομάζονται επαναλήψεις (Rasmusson, 2010).



Εικόνα 1: Agile Φιλοσοφία

Η Agile μεθοδολογία εξελίχθηκε στα μέσα της δεκαετίας του 1990 ως μέρος της αντίδρασης ενάντια στις παραδοσιακές μεθόδους καταρράκτη (*waterfall*). Σύμφωνα με τη μέθοδο αυτή η παράδοση του έργου γίνεται εξολοκλήρου κατά το πέρας του έργου. Το γεγονός ότι οι πρακτικές Agile προέρχονταν από τη χρήση του μοντέλου καταρράκτη τις καθιστούσε άκαμπτες, αργές και ασυνεπείς σχετικά με την αποτελεσματικότητα και την αποδοτικότητα των προγραμματιστών. Οι ευέλικτες μέθοδοι σηματοδοτούν μια επιστροφή στην πρακτική ανάπτυξης από τις αρχές της ιστορίας της ανάπτυξης λογισμικού. Η χρήση της λέξης agile σε αυτό το πλαίσιο προέρχεται από το agile μανιφέστο.



## 2.2 Agile Manifesto

Το Φεβρουάριο του 2001 στην Γιούτα των ΗΠΑ μια ομάδα 17 διακεκριμένων ερευνητών συμπεριέλαβε σε μορφή μανιφέστου τον επίσημο ορισμό για μια διαδικασία ανάπτυξης λογισμικού που να είναι ευέλικτη. Στη συνέχεια οι ερευνητές αυτοί παρουσίασαν σε ένα συνέδριο έναν αποδοτικό τρόπο ανάπτυξης λογισμικού. Μετά το συνέδριο δημιούργησαν την «*Agile Alliance*». (Agile Manifesto)

Όλες οι ευέλικτες μεθοδολογίες βασίζονται στις ίδιες βασικές αρχές και έχουν τον ίδιο στόχο, ωστόσο κάθε μία έχει διαφορετική προσέγγιση. (Sverrisdottir, Ingason, Jonasson, 2014). Στο Μανιφέστο δίνεται έμφαση στο πόσο σπουδαίο ρόλο παίζει η στενή και ομαλή συνεργασία μεταξύ των διαφόρων ομάδων συμμετοχών, όπως η ομάδα ανάπτυξης και όλοι οι ενδιαφερόμενοι φορείς. Η σημαντικότητα των μεθόδων έγκειται στην ικανότητα άμεσης ανταπόκρισης τους στις αλλαγές (Judy, Beens, 2008). Επιπλέον, ο ευέλικτος τρόπος σκέψης και δράσης είναι ζωτικής σημασίας προκειμένου ένα επιχειρηματικό πλάνο να στεφθεί με επιτυχία στην σημερινή εποχή. Η προσέγγιση αυτή μπορεί να εφαρμοστεί με επιτυχία σε έργα κάθε τύπου και όχι μόνο σε έργα πληροφορικής- ανάπτυξης λογισμικού καθώς το βασικό συστατικό είναι η άμεση ανατροφοδότηση. Η ευέλικτη φιλοσοφία είναι ολιστική και μπορεί να βρει εφαρμογές τόσο στον επιχειρηματικό κόσμο όσο και στην καθημερινή ζωή. Τα έργα που εμπεριέχουν αβεβαιότητα είναι συνετό να προσεγγίζονται με ευέλικτες πρακτικές (Stare, 2014).



Εικόνα 2: Agile Manifesto

### 2.2.1 Οι αξίες

#### 1. Τα άτομα και οι αλληλεπίδραση τους πάνω από τις διαδικασίες

Το agile δίνει μεγαλύτερη σημασία στα άτομα από αυτή που δίνει στη διαδικασία ακόμη και στα εργαλεία. Οι άνθρωποι ανταποκρίνονται στις επιχειρηματικές ανάγκες και καθοδηγούν τη διαδικασία ανάπτυξης. Οι διαδικασίες και τα εργαλεία από τη φύση τους είναι λιγότερο ευαίσθητα στην αλλαγή και δεν μπορούν να ανταποκριθούν από μόνα τους στις μεταβαλλόμενες ανάγκες των πελατών.

#### 2. Λειτουργία του λογισμικού βάσει ολοκληρωμένης τεκμηρίωσης

Η τεκμηρίωση απαιτεί χρόνο. Είναι μέρος μιας πιο αργής διαδικασίας στην οποία είναι ενάντια η μέθοδος agile. Κάθε τεκμηρίωση, από τεχνικές απαιτήσεις έως σχέδια δοκιμών και άλλες προδιαγραφές, απαιτεί έγκριση. Αυτό καθυστερεί την ανάπτυξη. Το Agile έχει ως στόχο τον εξ ορθολογισμό και όχι την εξάλειψη της τεκμηρίωσης. Ενώ η τεκμηρίωση έχει την αξία της, στην agile νοοτροπία, το λογισμικό είναι ύψιστης σημασίας.

#### 3. Συνεργασία σημαντικότερη από τις διαπραγματεύσεις με τους πελάτες

Οι πελάτες και οι διαχειριστές προϊόντων πρέπει να επεξεργαστούν τις λεπτομέρειες παράδοσης, αλλά αντί αυτού διαπραγματεύονται αυτήν τη διαδικασία και στο σημείο αυτό παρεμβαίνει η agile νοοτροπία. Για παράδειγμα, στις παραδοσιακές μεθοδολογίες διαχείρισης έργων, οι πελάτες διαπραγματεύονται λεπτομερώς τις απαιτήσεις για το προϊόν - πριν ξεκινήσει η εργασία. Ωστόσο, το agile συμπεριλαμβάνει τον πελάτη σε όλο το φάσμα της ζωής του έργου για τη συνεχή ανατροφοδότησή του.

#### 4. Η ανταπόκριση σε αλλαγές πάνω από τη τήρηση σε ένα πλάνο

Στη παραδοσιακή μεθοδολογία διαχείρισης έργου παρότι συμβαίνουν αλλαγές, αυτές πρέπει να αποφεύγονται όσο το δυνατόν περισσότερο. Αντ' αυτού, το agile λειτουργεί σε σύντομες επαναλήψεις που ονομάζονται σπριντ (sprint). Η συντομία τους επιτρέπει αλλαγές, ακόμη και τις αγκαλιάζει ως τρόπο βελτίωσης του έργου και αύξησης της προστιθέμενης αξίας (The agile manifesto explained).

### 2.2.2 Οι βασικές αρχές

Οι αρχές που διέπουν το Agile Manifest είναι οι εξής:

- Πρώτη προτεραιότητα είναι να ικανοποιήσουμε τον πελάτη παραδίδοντας του λειτουργικό λογισμικό διαρκώς και έγκαιρα.
- Ακόμη και αργά στην ανάπτυξη είναι καλοδεχούμενες οι αλλαγές. Οι ευέλικτες διαδικασίες αξιοποιούν την αλλαγή για το ανταγωνιστικό πλεονέκτημα του πελάτη.
- Παροχή λογισμικού εργασίας συχνά, από δύο εβδομάδες έως δύο μήνες, με προτίμηση στο μικρότερο χρονικό διάστημα.
- Οι επιχειρηματίες και οι προγραμματιστές πρέπει να συνεργάζονται στενά σε όλη τη διάρκεια του έργου.
- Δημιουργία έργων γύρω από άτομα με κίνητρα. Διαμόρφωση κατάλληλου περιβάλλοντος βασισμένα στην εμπιστοσύνη και την υποστήριξη όπου χρειάζεται για να ολοκληρωθεί η δουλειά.
- Η πιο αποδοτική και αποτελεσματική μέθοδος για τη μετάδοση πληροφορίας προς και εντός της ομάδας ανάπτυξης λογισμικού είναι η συνομιλία πρόσωπο με πρόσωπο.
- Το παραδοτέο λειτουργικό λογισμικό είναι το ουσιαστικό μέτρο για την πρόοδο.
- Οι ευέλικτες διαδικασίες προάγουν την αειφόρο ανάπτυξη. Οι χορηγοί, η ομάδα ανάπτυξης και οι τελικοί χρήστες θα πρέπει να διατηρούν ένα σταθερό ρυθμό επ' αόριστον.
- Η συνεχής προσοχή στην τεχνική αριστεία και ο καλός σχεδιασμός βελτιώνει την ευελιξία.
- Η απλότητα είναι ουσιώδης, όπως και η ικανότητα αντίληψης του όγκου της δουλειάς που δεν απαραίτητο να γίνει.
- Οι καλύτερες αρχιτεκτονικές, απαιτήσεις και σχέδια προκύπτουν από αυτο-οργανωμένες ομάδες.
- Σε τακτά χρονικά διαστήματα, η ομάδα σκέφτεται για το πώς θα γίνει πιο αποτελεσματική και ανάλογα ρυθμίζει και προσαρμόζει το τρόπο λειτουργίας της

Σύμφωνα με τις βασικές αρχές του μανιφέστου αναπτύχθηκαν όλες οι ευέλικτες μέθοδοι. Αρχικά, υπήρξε μια διακριτή στροφή (Dingsøyr, 2012) προς τη συνεργατική ανάπτυξη αναγνωρίζοντας ταυτόχρονα προνόμια στους ανθρώπους σε σχέση με τις διαδικασίες που δημιουργούσαν περιορισμούς παλιότερα. Με το πέρασμα των ετών, επικράτησε μια νοοτροπία λιτότητας προκειμένου να ελαχιστοποιηθούν οι περιττές εργασίες, κυρίως όσον αφορά τη δημιουργία τεκμηρίωσης, που πολλές φορές απαιτεί πολύ χρόνο. Επιπρόσθετα, οι πελάτες και άλλοι άμεσα ενδιαφερόμενοι άρχισαν να έχουν ενεργό συμμετοχή στη διαμόρφωση του τελικού προϊόντος. Τέλος, έγινε ξεκάθαρο το γεγονός ότι ένα αναπόσπαστο μέρος την εξέλιξη του έργου είναι η αβεβαιότητα που προκαλεί το διαρκώς μεταβαλλόμενο περιβάλλον. Αυτομάτως η προσπάθεια για έλεγχο και πρόβλεψη των απαραίτητων αλλαγών, ήταν μάταιη.

Οι αρχές της διαχείρισης έργου με agile προσέγγιση έχουν άμεση επίδραση στη δημιουργία προσαρμοσίμων προϊόντων, που είναι εύκολο να αλλάξουν με λιγότερο κόστος. Επίσης απαιτούνται ευπροσάρμοστες ομάδες έργου με δυνατότητα άμεσης ανταπόκρισης στις αλλαγές του περιβάλλοντος. Η ευέλικτη διαχείριση έργου, βασίζεται σε ευέλικτους υπαλλήλους, οι οποίοι λειτουργούν μέσα σε εταιρείες που είναι οι ίδιες ευέλικτες προκειμένου να μπορούν να προσαρμόζονται και να ανταποκρίνονται στις διαρκώς μεταβαλλόμενες απαιτήσεις του πελάτη και τις μεταβαλλόμενες συνθήκες του περιβάλλοντος (Γεωργιάδου, 2014).

### 2.3 Pros-Cons

Το Agile είναι ένα ισχυρό εργαλείο για την ανάπτυξη λογισμικού, όχι μόνο παρέχει οφέλη στην ομάδα ανάπτυξης, αλλά παρέχει επίσης ορισμένα σημαντικά επιχειρηματικά οφέλη στον πελάτη. Βοηθάει τις ομάδες έργων να αντιμετωπίσουν πολλές από τις πιο κοινές παγίδες του έργου (όπως κόστος, προβλεψιμότητα χρονοδιαγράμματος και λάθος κατανόηση του ζητούμενου αποτελέσματος) με πιο ελεγχόμενο τρόπο. Αναδιοργανώνοντας και επανεξετάζοντας τις δραστηριότητες που εμπλέκονται στην ανάπτυξη του προσαρμοσμένου προϊόντος, η μέθοδος Agile επιτυγχάνει τους ίδιους στόχους με πιο λιτό και πιο εστιασμένο τρόπο στις επιχειρήσεις.

### 2.3.1 Τα Θετικά

- **Διαφάνεια:** Δίνεται η δυνατότητα στον πελάτη να ελέγχει τα παραδοτέα σε προκαθορισμένο σύντομο χρονικό διάστημα και να λαμβάνει αποφάσεις για αλλαγές καθ' όλη τη διάρκεια του έργου.
- **Συμμετοχή ενδιαφερομένων:** Ο πελάτης αποκτά μια ισχυρή αίσθηση ιδιοκτησίας συνεργαζόμενος άμεσα με την ομάδα του έργου σε όλη τη διάρκεια ανάπτυξής του έργου. Με τη συμμετοχή του πελάτη σε κάθε βήμα του έργου, αναπτύσσεται ένας μεγάλος βαθμός συνεργασίας μεταξύ των συμμετεχόντων του έργου. Αυτό δίνει την δυνατότητα στην ομάδα ανάπτυξης να κατανοήσει ευκολότερα τις επιθυμίες του πελάτη σχετικά με το τελικό προϊόν - υπηρεσία. Η έγκαιρη παράδοση λειτουργικού αποτελέσματος αυξάνει την εμπιστοσύνη των ενδιαφερομένων στην ικανότητα της ομάδας ανάπτυξης και τους ενθαρρύνει να ασχοληθούν περισσότερο με το έργο.
- **Έγκαιρη και προβλέψιμη παράδοση:** Εάν ο χρόνος που απαιτείται για μια συγκεκριμένη εφαρμογή προκειμένου να κυκλοφορήσει στην αγορά είναι η μεγαλύτερη ανησυχία σε σχέση με το να κυκλοφορήσει ένα πλήρες σύνολο λειτουργιών κατά την αρχική εκκίνηση, το Agile μπορεί πιο γρήγορα να παράγει μια βασική έκδοση της λειτουργικής εφαρμογής που μπορεί να ολοκληρωθεί πλήρως σε διαδοχικές επαναλήψεις.
- **Βελτιωμένη ποιότητα:** Χωρίζοντας το έργο σε διαχειρίσιμες μονάδες, η ομάδα του έργου μπορεί να επικεντρωθεί στην ανάπτυξη, τη δοκιμή και τη συνεργασία υψηλής ποιότητας. Επίσης, παράγοντας συχνές κατασκευές και πραγματοποιώντας δοκιμές και κριτικές κατά τη διάρκεια κάθε επανάληψης, η ποιότητα βελτιώνεται με την εύρεση και διόρθωση ελαττωμάτων γρήγορα και τον εντοπισμό αναντιστοιχίας προσδοκιών νωρίς.
- **Η ανάπτυξη είναι συχνά πιο επικεντρωμένη στο χρήστη:** Το Agile χρησιμοποιεί συνήθως τις απαιτήσεις του πελάτη (*user stories*) με κριτήρια αποδοχής εστιασμένα στην επιχείρηση για να καθορίσει τις δυνατότητες του προϊόντος. Εστιάζοντας τις δυνατότητες στις ανάγκες που έχουν οι καθημερινοί χρήστες, κάθε δυνατότητα παρέχει σταδιακά αξία. Αυτό παρέχει επίσης την ευκαιρία να δοκιμαστεί μια beta έκδοση του προϊόντος μετά από κάθε Sprint, κερδίζοντας πολύτιμα σχόλια στις αρχές του έργου και παρέχοντας τη δυνατότητα να γίνουν αλλαγές ανάλογα με τις ανάγκες.

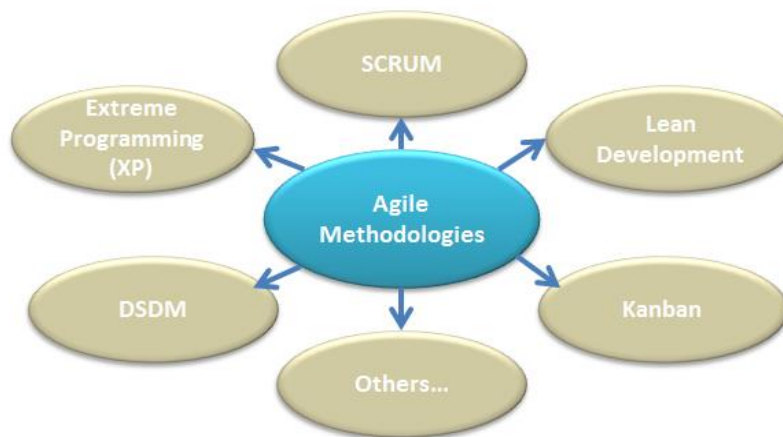
- **Προβλέψιμο Κόστος:** Επειδή κάθε Sprint έχει σταθερή διάρκεια, το κόστος είναι προβλέψιμο και περιορίζεται στο ποσό της εργασίας που μπορεί να εκτελέσει η ομάδα στο πλαίσιο καθορισμένου χρονοδιαγράμματος.

### 2.3.2 Τα Αρνητικά

- Οι πελάτες πρέπει να κατανοήσουν ότι βλέπουν ένα έργο σε εξέλιξη σε αντάλλαγμα για αυτό το πρόσθετο όφελος της διαφάνειας.
- Το Agile λειτουργεί καλύτερα όταν τα άτομα που απαρτίζουν την ομάδα ανάπτυξης είναι πλήρως αφοσιωμένα στο έργο.
- Επειδή το Agile επικεντρώνεται στην παράδοση με χρονικά διαστήματα και στη συχνή αναπροσαρμογή, είναι πιθανό ορισμένα στοιχεία που έχουν οριστεί για παράδοση να μην ολοκληρωθούν εντός του καθορισμένου χρονικού πλαισίου. Ίσως χρειαστούν επιπλέον χρονικές περίοδοι (*sprints*) επιπλέον από αυτές που είχαν υπολογιστεί αρχικά, αυξάνοντας το κόστος του έργου. Επιπλέον, η συμμετοχή των πελατών οδηγεί συχνά σε επιπρόσθετες απαιτήσεις κατά τη διάρκεια του έργου. Αυτό έχει σαν αποτέλεσμα και πάλι, την αύξηση του συνολικού χρόνου και του κόστους της εφαρμογής.
- Οι στενές εργασιακές σχέσεις σε ένα έργο Agile είναι πιο εύκολο να διαχειριστούν όταν τα μέλη της ομάδας βρίσκονται στον ίδιο φυσικό χώρο, κάτι που δεν είναι πάντα δυνατό. Ωστόσο, υπάρχουν διάφοροι τρόποι αντιμετώπισης αυτού του ζητήματος.
- Ο επαναληπτικός χαρακτήρας της ανάπτυξης του Agile μπορεί να οδηγήσει σε συχνή αναδιαμόρφωση αν το σύνολο των απαιτήσεων του πελάτη δεν ληφθεί υπόψη στον αρχικό σχεδιασμό της αρχιτεκτονικής. Χωρίς αυτήν την αναδιαμόρφωση, το τελικό προϊόν θα υπολείπεται της συνολικής απαιτούμενης ποιότητας. Αυτό γίνεται πιο έντονο σε εφαρμογές μεγαλύτερης κλίμακας ή σε συστήματα που περιλαμβάνουν υψηλό επίπεδο ολοκλήρωσης.

## 2.4 Οι πιο διαδεδομένες Agile πρακτικές- μέθοδοι

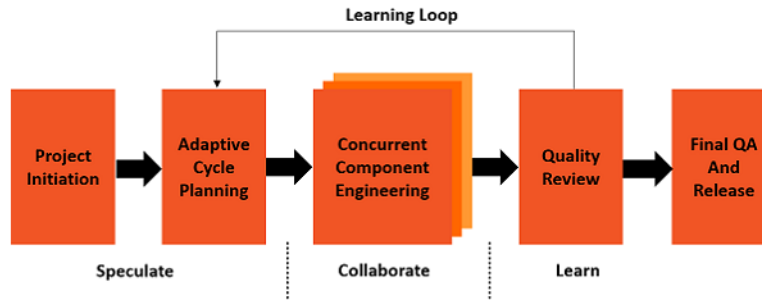
Πολλές agile μέθοδοι ανάπτυξης χρησιμοποιούνται ευρέως σήμερα. Μεταξύ τους διαφέρουν ως προς την προσέγγιση ανάπτυξης και τον τρόπο διαχείρισης που προτείνουν (Fowler, 2015). Ορισμένες επικεντρώνονται εκτενέστερα στις πρακτικές διαχείρισης έργων και συνεργασίας, όπως οι Scrum, Adaptive Software Development (ASD) και Lean Development. Ωστόσο, ορισμένες άλλες επικεντρώνονται σε μεγάλο βαθμό σε πρακτικές εφαρμογής λογισμικού όπως οι Extreme Programming (XP), Agile Modeling (AM) και Feature-driven Development (FDD).



Εικόνα 3: Agile Μεθοδολογίες

### 2.4.1 Adaptive software development (ASD)

Μεθοδολογία ανάπτυξης λογισμικού βασισμένη στην αρχή της συνεχούς προσαρμογής της διαδικασίας στο υπό εξέταση έργο. Η προσαρμοστική ανάπτυξη λογισμικού αντικαθιστά τον παραδοσιακό τρόπο καταρράκτη με μια επαναλαμβανόμενη σειρά κύκλων συνεργασίας και εκμάθησης. Παρέχει ένα πλαίσιο με επαρκή καθοδήγηση ώστε να αποτρέψει το έργο από το χάος, ταυτόχρονα όμως δεν περιορίζει την δημιουργική ελευθερία ανάπτυξης.



Εικόνα 4: Adaptive software development

Δομείται σε τρία στάδια: α) το στάδιο των εικασιών (Speculate), όπου κατά την έναρξη δημιουργείται το χρονοδιάγραμμα του έργου, αποφασίζεται ο αριθμός επαναλήψεων καθώς ο χρονικός τους ορίζοντας και τα χαρακτηριστικά τους μέσα από εικασίες. β) το στάδιο της συνεργασίας (Collaborate) στο οποίο γίνεται ταυτόχρονη ανάπτυξη χαρακτηριστικών και γ) το στάδιο της μάθησης - ανασκόπησης ποιότητας (Learn) κατά το οποίο δίνεται έμφαση στα αποτελέσματα από τεχνική άποψη, από τη πλευρά του πελάτη καθώς και στη λειτουργικότητα και τις πρακτικές που χρησιμοποιεί η ομάδα υλοποίησης. (ADS Practices)

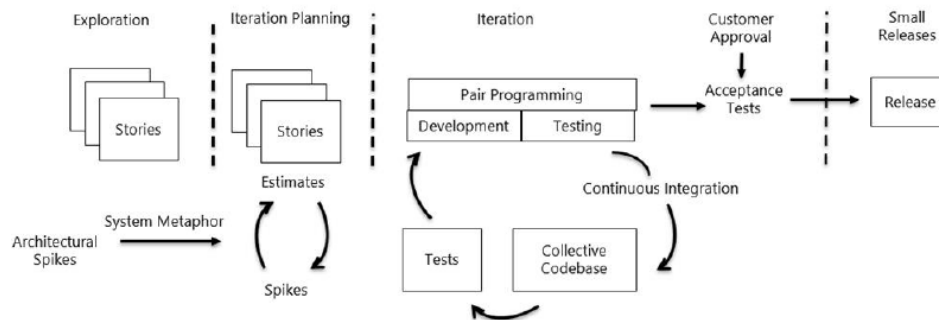
#### 2.4.2 Extreme Programming (XP)

Είναι ένα τυπικό πλαίσιο ανάπτυξης Agile, που αναπτύχθηκε από τον Kent Beck, και μπορεί να προσαρμοστεί σε αναπτυξιακές εταιρείες διαφόρων διαστάσεων. Πρόκειται για μια προσέγγιση που βασίζεται σε αξίες όπως είναι η Επικοινωνία, η Απλότητα, η Ανατροφοδότηση, το Θάρρος και ο Σεβασμός. Προτεραιότητα της είναι η ικανοποίηση του πελάτη ενώ ταυτόχρονα προσφέρει εμπιστοσύνη στους προγραμματιστές. Τους παρακινεί να αποδέχονται τις αλλαγές τις οποίες μπορεί να απαιτήσει ο πελάτης, ακόμη και αν το στάδιο ανάπτυξης του έργου βρίσκεται σε προχωρημένη φάση. Υποστηρίζει συχνές "εκδόσεις" σε σύντομους κύκλους ανάπτυξης, με σκοπό τη βελτίωση της αποδοτικότητας. Επιπλέον, σε κάθε κύκλο ανάπτυξης εισάγει σημεία ελέγχου στα οποία μπορούν να υιοθετηθούν νέες απαιτήσεις των πελατών. Άλλα στοιχεία ακραίου προγραμματισμού περιλαμβάνουν: προγραμματισμό σε ζευγάρια ή εκτεταμένη αναθεώρηση του κώδικα, δοκιμές όλου του κώδικα, αποφυγή προγραμματισμού χαρακτηριστικών έως ότου πραγματικά χρειαστούν, επίπεδη δομή διαχείρισης, απλότητα και σαφήνεια κώδικα, αναμένοντας



αλλαγές στις απαιτήσεις του πελάτη όσο το ζητούμενο είναι καλύτερα κατανοητό. Τέλος, συχνή επικοινωνία ανάμεσα στον πελάτη και στους προγραμματιστές.

## Extreme Programming (XP) at a Glance



Εικόνα 5: Extreme Programming (XP)

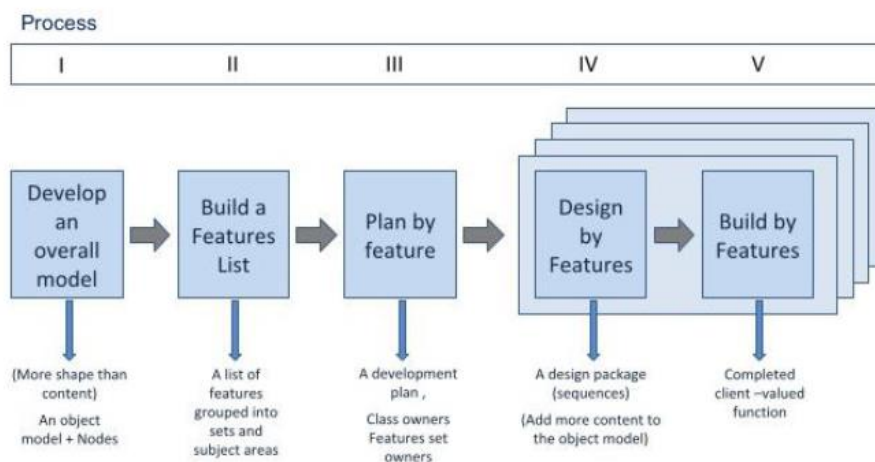
Γίνεται εύκολα αντιληπτό ότι η ομαδική εργασία είναι εξαιρετικά σημαντική σε αυτή τη μέθοδο, καθώς όταν παρουσιαστεί το οποιοδήποτε πρόβλημα, η επίλυση προέρχεται από το σύνολο όλων των εμπλεκόμενων φορέων.

### 2.4.3 Crystal Clear methods

Το Crystal Clear είναι μέλος της οικογένειας των μεθοδολογιών Crystal και θεωρείται ένα παράδειγμα ευέλικτης ή ελαφριάς μεθοδολογίας (Cockburn). Μπορεί να εφαρμοστεί σε ομάδες έως και 6 ή 8 προγραμματιστών. Η οικογένεια των μεθοδολογιών Crystal επικεντρώνεται στην αποδοτικότητα και τη βιωσιμότητα ως συστατικά της ασφάλειας του έργου. Το Crystal Clear εστιάζει σε ανθρώπους, όχι σε διαδικασίες ή αντικείμενα και απαιτεί συχνή παράδοση λειτουργικού κώδικα στους χρήστες, ανακλαστική βελτίωση και ωσμωτική επικοινωνία. Επιπλέον περιλαμβάνει τις εξής προαιρετικές ιδιότητες: εύκολη πρόσβαση σε εμπείρους χρήστες, προσωπική ασφάλεια και εστίαση. Τέλος, περιλαμβάνει αυτοματοποιημένες δοκιμές και συχνή ενσωμάτωση (Crystal clear methodology).

#### 2.4.4 Feature-driven development (FDD)

Είναι μια επαναληπτική και σταδιακή διαδικασία ανάπτυξης λογισμικού και αποτελεί μια από τις πολλές agile μεθόδους για την ανάπτυξη λογισμικού. Η FDD συνδυάζει τις βέλτιστες πρακτικές που αναγνωρίζονται από τη βιομηχανία σε ένα συνεκτικό σύνολο. Όλες αυτές οι πρακτικές καθοδηγούνται από την οπτική γωνία λειτουργικότητας των χαρακτηριστικών που δίνουν αξία στον πελάτη. Ο κύριος σκοπός της είναι να παρέχει απτό, λειτουργικό λογισμικό εγκαίρως. (Kumar & Bhatia, 2012).



Εικόνα 6: Feature-driven development (FDD)

#### 2.4.5 Dynamic systems development method (DSDM)

Είναι ένα ευέλικτο πλαίσιο παράδοσης έργου, που χρησιμοποιείται κυρίως ως μέθοδος ανάπτυξης λογισμικού. Αρχικά κυκλοφόρησε το 1994, προσπαθώντας να παρέχει πειθαρχία στη μέθοδο ταχείας ανάπτυξης εφαρμογών. Το 2007 έγινε μια γενική προσέγγιση για τη διαχείριση έργων και την παράδοση λύσεων. Το DSDM είναι μια επαναληπτική και σταδιακή προσέγγιση που βασίζεται στις αρχές της ανάπτυξης Agile, συμπεριλαμβανομένης της συνεχούς συμμετοχής χρηστών / πελατών. Το DSDM επιδιορθώνει το κόστος, την ποιότητα και τον χρόνο στην αρχή και χρησιμοποιεί την MoSCoW ιεράρχηση για να ορίσει προτεραιότητες και να προσαρμόσει το παραδοτέο έργο ανταποκρινόμενο στον καθορισμένο περιορισμό χρόνου. Το DSDM είναι μία από τις πολλές μεθόδους Agile για την ανάπτυξη λογισμικού και λύσεων εκτός IT, και αποτελεί μέρος της Agile Alliance.

#### 2.4.6 Lean Software Development (LSD)

Είναι μια επαναλαμβανόμενη μεθοδολογία Agile που εστιάζει την ομάδα στην παροχή αξίας στον πελάτη. Είναι μια εξαιρετικά ευέλικτη, εξελισσόμενη μεθοδολογία χωρίς αυστηρές οδηγίες, κανόνες ή μεθόδους. Οι βασικές αρχές της μεθοδολογίας Lean περιλαμβάνουν:

- Εξάλειψη των αποβλήτων
- Ενίσχυση της μάθησης
- Αποφάσεις όσο πιο αργά γίνεται
- Παράδοση όσο το δυνατόν γρηγορότερα
- Ενίσχυση της ομάδας
- Οικοδόμηση ακεραιότητας
- Οπτική του συνόλου



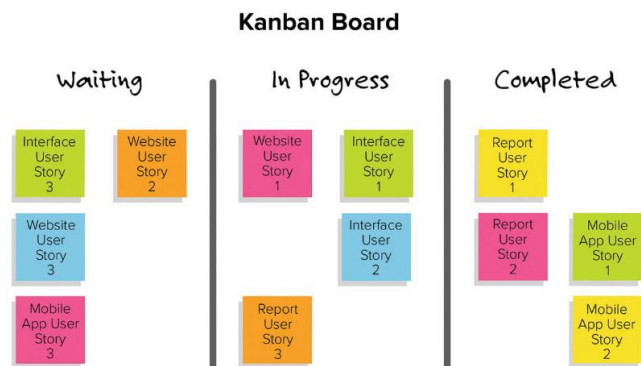
Εικόνα 7:Lean Software Development (LSD)

Η μέθοδος εξαλείφει τα περιττά χαρακτηριστικά ζητώντας από τους χρήστες να επιλέξουν μόνο τα πραγματικά πολύτιμα χαρακτηριστικά για ένα σύστημα, δίνοντας προτεραιότητα σε αυτά παραδίδοντάς τα σε μικρές παρτίδες. Βασίζεται σε γρήγορη και αξιόπιστη ανατροφοδότηση μεταξύ προγραμματιστών και πελατών, δίνοντας έμφαση στην ταχύτητα και την αποτελεσματικότητα των ροών εργασιών ανάπτυξης. Χρησιμοποιεί την ιδέα ενός προϊόντος εργασίας να «τραβιέται» μέσω του αιτήματος του πελάτη. Παρέχει εξουσία λήψης αποφάσεων σε

άτομα και μικρές ομάδες, δεδομένου ότι έχει αποδειχθεί ότι είναι μια ταχύτερη και πιο αποτελεσματική μέθοδος από μια ιεραρχική ροή ελέγχου. Επικεντρώνεται επίσης στην αποτελεσματική χρήση των πόρων της ομάδας, προσπαθώντας να διασφαλίσει ότι όλοι είναι όσο το δυνατόν πιο παραγωγικοί για το μέγιστο χρονικό διάστημα. Συνιστά ανεπιφύλακτα να γράφονται αυτοματοποιημένες διαδικασίες ελέγχου ταυτόχρονα με τον κώδικα.

#### 2.4.7 Kanban

Το Kanban (πινακίδα στα ιαπωνικά) είναι ένα σύστημα προγραμματισμού για lean development και κατασκευή just-in-time (JIT) (Ohno, 1998). Αποτελεί ένα σύστημα ελέγχου απογραφής για τον έλεγχο της αλυσίδας εφοδιασμού. Ο Taiichi Ohno, μηχανικός της Toyota, ανέπτυξε το Kanban για να βελτιώσει την αποδοτικότητα της κατασκευής. Το Kanban είναι μια μέθοδος για την επίτευξη του JIT.



*Εικόνα 8: Kanban Board example*

Το Kanban έγινε ένα αποτελεσματικό εργαλείο για να υποστηρίξει τη λειτουργία ενός συστήματος παραγωγής και έναν εξαιρετικό τρόπο προώθησης της βελτίωσης. Ένα από τα κύρια πλεονεκτήματα του Kanban είναι η καθιέρωση ανώτατου ορίου στην απογραφή των εργασιών, αποφεύγοντας την υπερφόρτωση του κατασκευαστικού συστήματος. Ένας αγγλικός όρος αποτυπώνει την έννοια της ιαπωνικής λέξης, Kanban, ως έναν περιοριστή ουράς αναμονής (Waldner, 1992). Καθώς αντιμετωπίζονται τα προβλήματα της διαδικασίας, το όριο ουράς (ή το μέγιστο) μειώνονται.

#### 2.4.8 Scrum

Είναι μια μέθοδος που προτάθηκε ως ένας πιο αποτελεσματικός και ευέλικτος τρόπος για την ανάπτυξη έργων λογισμικού και ως εναλλακτική λύση σε άλλες μεθοδολογίες όπως το Waterfall, όπου ένα έργο σχεδιάζεται στην αρχή και ολοκληρώνεται σε διαφορετικά στάδια έως ότου κυκλοφορήσει στους πελάτες του (Sverrisdottir, 2014).

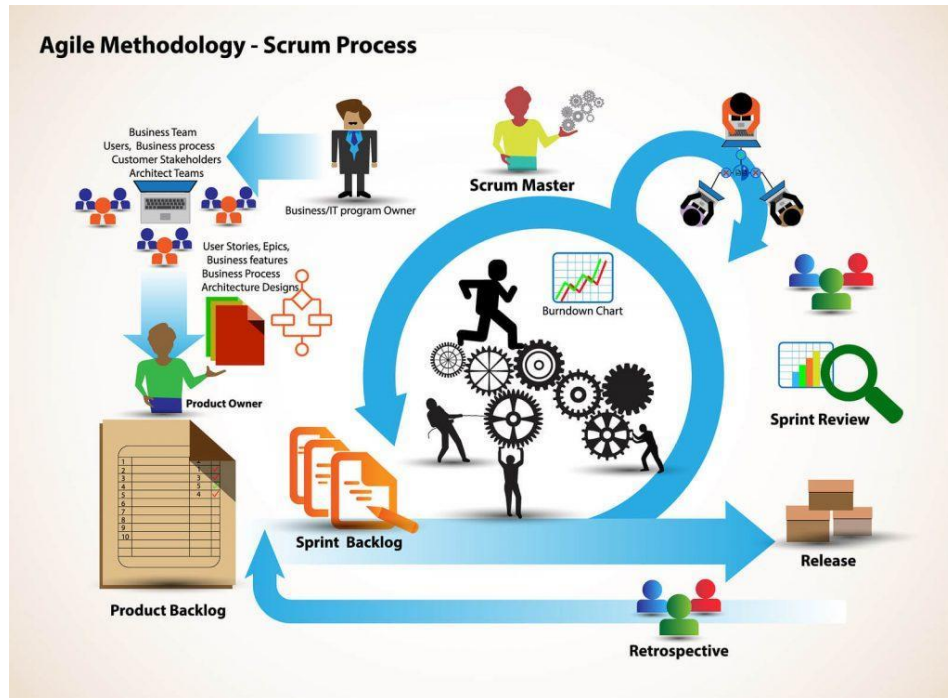
Βασίζεται στην ακόλουθη ιδέα *"Κάθε φορά που ξεκινάει ένα έργο, γιατί να μην γίνεται συχνά ένας έλεγχος προκειμένου να διαπιστωθεί αν το αποτέλεσμα κατευθύνεται προς τη σωστή κατεύθυνση και εάν είναι πραγματικά αυτό που θέλουν οι πελάτες;"*. Το Scrum βασίζεται σε εμπειρικές διαδικασίες, δηλαδή βασίζεται στην υπόθεση ότι είναι δύσκολο να προβλέψουμε και να σχεδιάσουμε ένα έργο από την αρχή έως το τέλος γι' αυτό είναι πιο αποτελεσματικό ανά τακτά χρονικά διαστήματα να ελέγχουμε τη πρόοδο. Στο βιβλίο *"Ευέλικτη Ανάπτυξη λογισμικού με την Scrum"* περιγράφεται και αναλύεται με λεπτομέρεια η μέθοδος. (Schwaber, Beedle, 2001).

Η συγκεκριμένη μέθοδος εμφανίζει επιτυχία που οφείλεται στον τρόπο προσέγγισης και διαχείρισης του λογισμικού. Προσδίδει ευελιξία στις διαδικασίες οργάνωσης και διαχείρισης κατά την ανάπτυξη λογισμικού. Ταυτόχρονα, σε αυτή οφείλεται η αύξηση της παραγωγικότητας των μελών της ομάδας καθώς και η προσαρμοστικότητα τους ανάλογα με τις ανάγκες που δημιουργούνται. Τέλος, δίνεται έμφαση στην επίτευξη της αύξησης της αποτελεσματικότητας της ομάδας καθώς το περιβάλλον μεταβάλλεται διαρκώς.

Η προσέγγιση που χρησιμοποιείται στο Scrum για το σχεδιασμό και τη διαχείριση της εργασίας είναι μέσω επαναλήψεων διάρκειας μεταξύ δύο και τεσσάρων εβδομάδων. Αυτές οι επαναλήψεις είναι γνωστές ως Sprints και εκεί αναπτύσσεται η προγραμματισμένη εργασία και όπου πραγματοποιούνται οι συναντήσεις που σχετίζονται με το Scrum. Τα σπριντ γίνονται διαδοχικά επιτρέποντας την ανάπτυξη της εργασίας σε επαναληπτική και σταδιακή προσέγγιση έως ότου το προϊόν είναι έτοιμο προς παράδοση. Όταν ολοκληρωθεί το Sprint, το ακόλουθο Sprint ξεκινά αμέσως.

Η Μέθοδος Scrum απαρτίζεται από πέντε σημαντικά στάδια:

1. Απαιτήσεις
2. Σχεδιασμός
3. Ανάπτυξη
4. Ολοκλήρωση & δοκιμές
5. Εφαρμογή & Ανάπτυξη
6. Αναθεώρηση



Εικόνα 9: Scrum Process

- **Απαιτήσεις**

Προτού ο product owner είναι σε θέση να σχεδιάζει ένα έργο, πρέπει να δημιουργήσει την αρχική τεκμηρίωση που θα απαριθμεί τις αρχικές απαιτήσεις. Αυτές είναι: το τελικό αποτέλεσμα που θα επιτύχει το έργο, τα χαρακτηριστικά που θα υποστηρίζει, τα χαρακτηριστικά που δεν υποστηρίζει αρχικά. Καλό είναι οι αρχικές απαιτήσεις να είναι όσο το δυνατόν λιγότερες, προσθέτοντας μόνο τα σίγουρα απαραίτητα χαρακτηριστικά και αγνοώντας αυτά που δεν θα χρησιμοποιούνται συχνά. Οι προγραμματιστές μπορούν να εργαστούν σε αυτά αργότερα, μόλις αναπτυχθεί η εφαρμογή και οι βασικές λειτουργίες λειτουργούν καλά. Σε περαιτέρω επαναλήψεις, ο πελάτης και ο product owner μπορούν να εξετάσουν τις απαιτήσεις και τις καταστήσουν πιο σχετικές.

- **Σχεδιασμός**

Υπάρχουν δύο τρόποι προσέγγισης του σχεδιασμού στην ανάπτυξη λογισμικού - ο ένας είναι ο οπτικός σχεδιασμός και ο άλλος είναι η αρχιτεκτονική δομή της εφαρμογής.

#### Σχεδιασμός λογισμικού

Κατά την πρώτη επανάληψη, ο product owner συγκεντρώνει την ομάδα ανάπτυξης και εισάγει τις απαιτήσεις που δημιουργήθηκαν κατά το προηγούμενο στάδιο. Στη συνέχεια, η ομάδα συζητά πώς να αντιμετωπίσει αυτές τις απαιτήσεις και προτείνει τα εργαλεία που απαιτούνται για την επίτευξη του καλύτερου αποτελέσματος. Για παράδειγμα, η ομάδα καθορίζει τη γλώσσα προγραμματισμού, τα πλαίσια και τις βιβλιοθήκες που πρόκειται να χρησιμοποιήσει το έργο. Σε περαιτέρω επαναλήψεις, οι προγραμματιστές συζητούν την υλοποίηση των δυνατοτήτων.

#### Σχεδιασμός UI / UX

Κατά τη διάρκεια αυτού του σταδίου του κύκλου ζωής ανάπτυξης λογισμικού, οι σχεδιαστές δημιουργούν μια πρόχειρη μακέτα του UI. Εάν το προϊόν απευθύνεται σε καταναλωτές, η διεπαφή χρήστη και η εμπειρία χρήστη είναι πιο σημαντικές. Περαιτέρω επαναλήψεις δαπανώνται για τον εξευγενισμό του αρχικού σχεδιασμού και / ή την επανεπεξεργασία του ώστε να ταιριάζει με τα νέα χαρακτηριστικά.

- **Ανάπτυξη**

Η φάση ανάπτυξης αφορά τη σύνταξη κώδικα και τη μετατροπή της τεκμηρίωσης σχεδιασμού σε πραγματικό λογισμικό κατά τη διαδικασία ανάπτυξης λογισμικού. Αυτό το στάδιο του κύκλου ζωής ανάπτυξης λογισμικού είναι γενικά το μεγαλύτερο, καθώς αποτελεί τη ραχοκοκαλιά της όλης διαδικασίας. Δεν υπάρχουν πολλές αλλαγές μεταξύ των επαναλήψεων στο στάδιο αυτό.

- **Ολοκλήρωση & δοκιμές**

Αυτό το στάδιο δαπανάται για να εξασφαλισθεί ότι το λογισμικό είναι χωρίς σφάλματα και συμβατό με όλα τα άλλα που έχουν γράψει οι προγραμματιστές στο παρελθόν. Η ομάδα διασφάλισης ποιότητας διεξάγει μια σειρά δοκιμών προκειμένου να διασφαλίσει ότι ο κώδικας είναι καθαρός και ότι επιτυγχάνονται οι επιχειρηματικοί στόχοι της λύσης.

Κατά τη διάρκεια των περαιτέρω επαναλήψεων αυτού του σταδίου του κύκλου ζωής ανάπτυξης λογισμικού, ο έλεγχος γίνεται περισσότερο εμπλεκόμενος και λαμβάνει υπόψη όχι μόνο τη δοκιμή λειτουργικότητας, αλλά και την ολοκλήρωση συστημάτων, τη διαλειτουργικότητα και τον έλεγχο αποδοχής χρηστών κ.λπ.

- **Εφαρμογή & ανάπτυξη**

Η εφαρμογή αναπτύσσεται στους διακομιστές και παρέχεται στους πελάτες - είτε για την επίδειξη είτε για την πραγματική χρήση. Περαιτέρω επαναλήψεις ενημερώνουν το ήδη εγκατεστημένο λογισμικό, εισάγοντας νέες δυνατότητες και επιλύοντας σφάλματα.

- **Αναθεώρηση**

Μόλις ολοκληρωθούν όλες οι προηγούμενες φάσεις ανάπτυξης, ο product owner συγκεντρώνει την ομάδα ανάπτυξης για άλλη μια φορά και εξετάζει την πρόοδο που έχει σημειωθεί για την ολοκλήρωση των απαιτήσεων. Η ομάδα εισάγει τις ιδέες της για την επίλυση των προβλημάτων που προέκυψαν κατά τις προηγούμενες φάσεις και ο product owner λαμβάνει υπόψη τις προτάσεις της.

#### *2.4.8.1 Βασικές Οντότητες*

- **Διαχειριστής (Scrum Master)**

Είναι υπεύθυνος για την ομαλή λειτουργία της ομάδας ενώ ταυτόχρονα φροντίζει την ορθή ανάπτυξη του έργου σύμφωνα με τους κανόνες και τις προδιαγραφές που έχουν καθοριστεί. Αλληλοεπιδρά τόσο με την ομάδα ανάπτυξης όσο και με τον πελάτη κατά τη διάρκεια της υλοποίησης. Καθήκον του είναι επίσης η επίλυση κάθε προβλήματος εντός της ομάδας προκειμένου η παραγωγικότητα να είναι όσο το δυνατόν υψηλότερη.

- **Ιδιοκτήτης του προϊόντος (Product Owner)**

Είναι ο υπεύθυνος διαχείρισης και ελέγχου του έργου, ενώ καταγράφει και τις απαιτήσεις. Επιλέγεται από τον διαχειριστή, τον πελάτη και τη διοίκηση. Είναι αυτός που επιλέγει σε συνεργασία με την ομάδα ανάπτυξης ποιες απαιτήσεις θα υλοποιηθούν σε κάθε επανάληψη και με ποια σειρά. Συμβάλλει στην εκτίμηση της προσπάθειας που θα χρειαστεί για την υλοποίηση, και ταυτόχρονα βρίσκει λύσεις σε τυχόν προβλήματα κατά τη διάρκεια της ανάπτυξης.



- Ομάδα ανάπτυξης (Scrum Team)

Η ομάδα Scrum είναι ένα σύνολο ατόμων (συνήθως μεταξύ πέντε και εννέα μελών) που συνεργάζονται για να παραδώσουν τις απαιτούμενες και προσυμφωνημένες προσθήκες του προϊόντος. Το πλαίσιο Scrum ενθαρρύνει ένα υψηλό επίπεδο επικοινωνίας μεταξύ των μελών της ομάδας, έτσι ώστε η ομάδα να μπορεί ακολουθήστε έναν κοινό στόχο, να τηρούνται οι ίδιοι κανόνες και να υπάρχει σεβασμός μεταξύ των μελών. Είναι δομημένη και εξουσιοδοτημένη να οργανώνει και να διαχειρίζεται την εργασία που της αναλογεί. Η προκύπτουσα συνεργασία βελτιστοποιεί τη συνολική αποδοτικότητα και αποτελεσματικότητα της. Το Scrum δεν αναγνωρίζει υπο-ομάδες στην ομάδα ανάπτυξης, ανεξάρτητα από τομείς που πρέπει να αντιμετωπιστούν όπως δοκιμές, αρχιτεκτονική, λειτουργίες ή επιχειρηματικές αναλύσεις. Τα μέλη της ομάδας ανάπτυξης μπορεί να έχουν εξειδικευμένες δεξιότητες και τομείς εστίασης, αλλά η αποτελεσματικότητα λαμβάνεται υπόψη στο σύνολο της ομάδας σας αποτέλεσμα συνεργασίας.

- Πελάτης

Έχει ενεργή συμμετοχή σε όλη την διαδικασία ανάπτυξης του έργου, ιδιαίτερα στις διαδικασίες καταγραφής, επιλογής και ανάπτυξης των απαιτήσεων, αλλά και στον έλεγχο καλής λειτουργίας των επιμέρους παραδοτέων, κατά συνέπεια και συνολικά του τελικού προϊόντος. Όταν ο πελάτης είναι και ο τελικός χρήστης του προϊόντος – υπηρεσίας η συμβολή του είναι ακόμα πιο σημαντική καθώς μπορεί να εντοπίσει σφάλματα και να ζητήσει αλλαγές ή και βελτιώσεις.

- Διοίκηση

Τα μέλη της διοίκησης έχουν στόχο να παρέχουν το οργανωτικό όραμα. Το όραμα είναι σημαντικό για τα μέλη της ομάδας ώστε να μπορούν να συσχετίζουν τις καθημερινές τους ενέργειες. Επίσης, κύριος ρόλος των μελών της διοίκησης είναι να αφαιρούν εμπόδια τα οποία δεν είναι στην αρμοδιότητα του scrum master. Επιπλέον, βοηθούν στην ατομική ανάπτυξη, καθώς παρέχουν καθοδήγηση σε επαγγελματικό αλλά ορισμένες φορές και σε προσωπικό επίπεδο. Τέλος προκειμένου να αποφευχθεί η στασιμότητα στην ανάπτυξη και την εξέλιξη των ομάδων είναι

απαραίτητη κάποιου είδους πρόκληση, την οποία η διοίκηση οφείλει να παρέχει. Scrum (Schwaber & Sutherland, 2013).

#### 2.4.8.2 Πρακτικές

- Κατάλογος ανεκτέλεστων απαιτήσεων (Product backlog)

Είναι μια λίστα που περιέχει τις γνωστές, μέχρι την παρούσα περίοδο, απαιτήσεις του συστήματος. Οι απαιτήσεις αυτές προέρχονται από τον πελάτη, ή τα τμήματα πωλήσεων, μάρκετινγκ, υποστήριξης πελατών ή τους ίδιους τους υπεύθυνους για την ανάπτυξη λογισμικού. Ο product owner είναι υπεύθυνος για την ταξινόμηση των θεμάτων σύμφωνα με την προτεραιότητα τους. Σε κάθε sprint η λίστα ενημερώνεται, τα ολοκληρωμένα θέματα αφαιρούνται ενώ προστίθενται νέα στοιχεία.

- Εκτίμηση προσπάθειας

Είναι μια διαδικασία που επαναλαμβάνεται προκειμένου να εκτιμηθούν με περισσότερη ακρίβεια οι διαθέσιμες πληροφορίες. Επομένως όσο αυξάνονται οι πληροφορίες για τις απαιτούμενες εργασίες τόσο καλύτερη γίνεται η αξιολόγηση του φόρτου εργασίας που θα απαιτηθεί. Στην διαδικασία αυτή συμμετέχουν ο product owner και η ομάδα υλοποίησης.

- Κύκλος επανάληψης (sprint)

Είναι μια επανάληψη ενός χρονικού πλαισίου ενός κύκλου συνεχούς ανάπτυξης. Μέσα σε ένα Sprint, η προγραμματισμένη ποσότητα εργασίας πρέπει να ολοκληρωθεί από την ομάδα ανάπτυξης και να είναι έτοιμη για έλεγχο. Οι ομάδες συνήθως ορίζουν μια σύντομη διάρκεια ενός σπριντ έως 2-4 εβδομάδες. Η ομάδα θέτει συνεργατικά τον στόχο της με τον product owner ως "Sprint Goal" και σχεδιάζει τη δουλειά τους στο "Sprint backlog" Μόλις ξεκινήσει το Sprint μετά τον προγραμματισμό της περιόδου, η ομάδα συνεργάζεται για να ολοκληρώσει αποτελεσματικά την προγραμματισμένη εργασία και να την προετοιμάσει για έλεγχο μέχρι το τέλος αυτής της περιόδου.

- Συνάντηση για το σχεδιασμό της επανάληψης (sprint planning meeting)

Με τη συνάντηση αυτή ξεκινά το Sprint σχεδιάζοντας τις εργασίες που πρέπει να εκτελεστούν κατά τη διάρκεια του. Το πλάνο που δημιουργείται κατά τη συνάντηση αυτή είναι αποτέλεσμα συνεργατικής εργασίας ολόκληρης της Scrum ομάδας. Ο product owner προτείνει πώς το προϊόν θα μπορούσε να αυξήσει την αξία και τη χρησιμότητά του στο τρέχον Sprint. Στη συνέχεια, ολόκληρη η ομάδα Scrum συνεργάζεται για να καθορίσει έναν στόχο. Μέσα από συζήτηση με τον product owner, τα μέλη της ομάδας επιλέγουν στοιχεία από το Product Backlog που θα συμπεριληφθούν στο τρέχον Sprint.. Αυτό γίνεται συχνά αποσυνθέτοντας μεγάλες απαιτήσεις σε μικρότερα αντικείμενα εργασίας μίας ημέρας ή λιγότερο. Το πώς γίνεται αυτό είναι κατά την αποκλειστική κρίση των προγραμματιστών.

- Λίστα απαιτήσεων επανάληψης (sprint backlog)

Η λίστα αυτή περιέχει ένα σύνολο στοιχείων προϊόντος που έχουν επιλεγεί να υλοποιηθούν για τη τρέχον Sprint. Μόνο η ομάδα ανάπτυξης μπορεί να τροποποιήσει το Sprint Backlog, το οποίο είναι εξαιρετικά ξεκάθαρο και αντικατοπτρίζει σε πραγματικό χρόνο τα σχέδια της ομάδας να ολοκληρώσει τις εργασίες της στο τρέχον Sprint. Μέσω της λίστας διευκρινίζεται το έργο που απαιτείται από την ομάδα ανάπτυξης για την επίτευξη των στόχων του Sprint. Όταν εμφανιστεί μια νέα δουλειά, η ομάδα ανάπτυξης πρέπει να την προσθέσει στη λίστα υποχρεώσεων του Sprint. Καθώς οι εργασίες προχωρούν ή ολοκληρώνονται, ο εκτιμώμενος υπολειπόμενος φόρτος εργασίας για κάθε εργασία ενημερώνεται.

- Καθημερινή συνάντηση ομάδας (daily meeting)

Μικρής διάρκειας συνάντηση που πραγματοποιείται σε καθημερινή βάση. Οι συσκέψεις πραγματοποιούνται συνήθως στην ίδια τοποθεσία και την ίδια ώρα κάθε μέρα. Στην ιδανική περίπτωση, μια καθημερινή συνάντηση scrum πραγματοποιείται το πρωί, καθώς συμβάλλει στον καθορισμό του πλαισίου για την εργασία της επόμενης ημέρας. Αυτές οι συσκέψεις scrum είναι αυστηρά διάρκειας μέχρι 15 λεπτά. Κατά τη διάρκεια της συνάντησης, κάθε μέλος της ομάδας απαντά στις ακόλουθες τρεις ερωτήσεις: Τι έκανε χθες, Τι θα κάνει σήμερα και αν έχει κάποιο εμπόδιο. Εστιάζοντας σε αυτό που κάθε άτομο πέτυχε χθες και θα πετύχει σήμερα, η ομάδα

αποκτά μια εξαιρετική κατανόηση του τι έχει γίνει και τι δουλειά μένει. Είναι μια συνάντηση στην οποία τα μέλη της ομάδας δεσμεύονται μεταξύ τους.

- Επισκόπηση στο τέλος της επανάληψης (sprint review)

Ο σκοπός της επισκόπησης Sprint είναι να επιθεωρήσει τα παραδοτέα που ολοκληρώθηκαν στο τέλος της επανάληψης και να καθορίσει πιθανές μελλοντικές προσαρμογές. Η Ομάδα Scrum παρουσιάζει τα αποτελέσματα της δουλειάς της στους βασικούς ενδιαφερόμενους και συζητείται η πρόοδος προς τον στόχο του προϊόντος. Κατά τη διάρκεια της συνάντησης, η Ομάδα Scrum και οι ενδιαφερόμενοι μελετούν τι επιτεύχθηκε στο Sprint και τι έχει αλλάξει στο περιβάλλον τους. Αξιολογείται η ανάπτυξη του έργου ενώ με βάση αυτές τις πληροφορίες, οι συμμετέχοντες συνεργάζονται για το τι πρέπει να κάνουν στη συνέχεια. Το Product Backlog μπορεί επίσης να προσαρμοστεί ώστε να ανταποκρίνεται σε νέες ευκαιρίες. (Schwaber & Sutherland, 2013).

- Συνεδρίαση Retrospective

Είναι μια συνάντηση που πραγματοποιείται στο τέλος μιας επανάληψης στην ανάπτυξη λογισμικού. Κατά τη διάρκεια της συνάντησης, η ομάδα σκέφτεται τι συνέβη στην επανάληψη και προσδιορίζει τις ενέργειες για τη βελτίωση που θα ακολουθήσει. Κάθε μέλος της ομάδας απαντά στις ακόλουθες ερωτήσεις: Τί λειτούργησε καλά εντός της ομάδας, τί δεν λειτούργησε καλά και τί ενέργειες πρέπει να γίνουν ώστε να βελτιωθεί η διαδικασία ανάπτυξης.

Η αναδρομική προοπτική του Agile μπορεί να θεωρηθεί ως συνάντηση «διδασμάτων». Η ομάδα σκέφτεται πώς πήγαν τα πάντα και στη συνέχεια αποφασίζει ποιες αλλαγές θέλουν να κάνουν στην επόμενη επανάληψη. Η αναδρομική καθοδήγηση καθοδηγείται από την ομάδα και τα μέλη της ομάδας πρέπει να αποφασίσουν μαζί πώς θα διεξαχθούν οι συναντήσεις και πώς θα ληφθούν αποφάσεις για βελτιώσεις.

## Κεφάλαιο 3<sup>ο</sup>

### 3.1 Μελέτη περίπτωσης: Softone Technologies SA

Η Softone Technologies A.E. είναι εταιρεία πληροφορικής με ηγετική της θέση στην αγορά. Δραστηριοποιείται στον κλάδο των συστημάτων ERP και CRM. Ιδρύθηκε το 2002 στην Αθήνα και έκτοτε έχει λάβει διακρίσεις και βραβεία για την καινοτομία των προϊόντων και των λύσεων που παρέχει στους πελάτες της

Το πελατολόγιο της απαρτίζεται κυρίως από μικρές και μεσαίες επιχειρήσεις αλλά πλέον εξυπηρετεί και τις σύνθετες ανάγκες πιο μεγάλων επιχειρήσεων προσφέροντας λύσεις μηχανογράφησης με δύο βασικά μοντέλα on-premise ή Software as a Service (SaaS)

Στο δυναμικό της εταιρείας συγκαταλέγονται έμπειρα στελέχη του χώρου. Απασχολούνται περισσότεροι από 170 εργαζόμενοι συνολικά. Η εταιρεία διαθέτει γραφεία και στη Θεσσαλονίκη, Ταυτόχρονα στο εξωτερικό δραστηριοποιούνται οι θυγατρικές εταιρείες σε Ρουμανία, Κύπρο και Βουλγαρία για την εξυπηρέτηση των συμφερόντων της εταιρείας στην ευρύτερη αγορά της ΝΑ Ευρώπης.

Το πελατολόγιο της υπερβαίνει τις 20.000 επιχειρήσεις και μέσα μέσα από ένα ισχυρό δίκτυο - περισσότερων των 300 - εξειδικευμένων συνεργατών, προσφέροντας στους πελάτες της μια πληθώρα λύσεων λογισμικού, που ενσωματώνουν όλες τις επιχειρησιακές τους λειτουργίες σε μία πλατφόρμα ικανή και ισχυρή.

Το 2007, παρουσίασε πρώτη σύγχρονες υπηρεσίες μηχανογράφησης βασισμένες στο διεθνές μοντέλο SaaS (Software as Service). Σήμερα κατέχει την ηγετική θέση στην εγχώρια αγορά cloud ERP. Περισσότερες από 4.500 επιτυχημένες εγκαταστάσεις λειτουργούν βασισμένες στο μοντέλο αυτό. Το 2011, διακρίθηκε στον παγκόσμιο διαγωνισμό αναδυόμενων τεχνολογιών της Microsoft, για τη λειτουργία του Soft1 ERP στην cloud πλατφόρμα Microsoft Windows Azure.

Το 2013, παρουσίασε στην αγορά την πρωτοποριακή έκδοση λογισμικού Soft1 Open Enterprise που αλλάζει τα δεδομένα στο μοντέλο λειτουργίας των επιχειρήσεων, συνδυάζοντας πλήρως τα οφέλη του cloud με τις προηγμένες λύσεις enterprise mobility της εταιρείας.

Η Softone εφαρμόζει από το 2014, ολοκληρωμένα Συστήματα Διαχείρισης Υπηρεσιών Πληροφορικής και Ασφάλειας Πληροφοριών, βασισμένα στον διεθνή αναγνωρισμένο πρότυπα ISO/IEC 20000:1-2011 και EN ISO 27001:2013 αντίστοιχα, Υιοθετώντας πρακτικές που διασφαλίζουν ποιότητα και αξιοπιστία στις παρεχόμενες υπηρεσίες της.

Η Softone επιδιώκει και έχει θέση ως προτεραιότητα να έχει πρωταγωνιστικό ρόλο στις εξελίξεις που αφορούν το cloud, στη συστηματική ανάπτυξη καινοτόμων λύσεων με σκοπό να παρέχει στις επιχειρήσεις υψηλή προστιθέμενη αξία, σε χαμηλό κόστος .

### 3.2 Εταιρική δομή

Το οργανωτικό σχήμα της Softone όπως έχει διαμορφωθεί διευκολύνει τη λήψη και την υλοποίηση αποφάσεων στο πλαίσιο εφαρμογής της στρατηγικής της εταιρείας. Μέσα από το οργανόγραμμα της εταιρείας καθορίζονται οι ρόλοι και οι σχέσεις που προκύπτουν για το συντονισμό της παραγωγής - ανάπτυξης τόσο του βασικού προϊόντος «ERP» όσο και των επιμέρους βοηθητικών προγραμμάτων με το κεντρικό σύστημα ελέγχου για όλη την εταιρεία. Ορισμένα από τα πιο βασικά τμήματα που σχετίζονται αμιγώς με το προϊόν στην εταιρεία είναι αυτό της διαχείρισης προϊόντος (*Product management*), τα τμήματα παραγωγής - έρευνας & ανάπτυξης (*R&D*), το τμήμα υποστήριξης πελατών (*Support*) καθώς και τμήματα συντονισμού και υλοποίησης έργων.

- Τμήμα διαχείριση προϊόντος

Σκοπός του τμήματος είναι η πλήρης επίβλεψη και ο συνολικός έλεγχος του τελικού προϊόντος. Έχει γνώση για τις ανάγκες τις αγορές, τις προβλεπόμενες νομοθεσίες που μεταβάλλονται συχνά καθώς και τη θέση του προϊόντος σε σχέση με τον ανταγωνισμό. Βασική αρμοδιότητα του τμήματος είναι η συνεχής συλλογή των νέων απαιτήσεων, η πληροφόρηση σχετικά με την ισχύουσα νομοθεσία καθώς και η επαλήθευση των ελαττωμάτων «*bugs*» που αναφέρονται είτε από τους πελάτες είτε από άλλα τμήματα εντός της εταιρείας.

Στο τμήμα αυτό απαιτείται η βαθιά γνώση του συνόλου του προγράμματος και η επιχειρησιακή γνώση «*business logic*». Τόσο οι γνώσεις λογιστικής, χρηματοοικονομικών, εφοδιαστικής αλυσίδας, κοστολόγησης όσο και η τεχνική κατάρτιση καθώς και η

συνεργασία με τα τμήματα έρευνας και ανάπτυξης είναι άμεση και χρειάζεται κοινή γλώσσα επικοινωνίας.

- Τμήματα Έρευνας & Ανάπτυξης

Τμήματα που αποτελούνται αποκλειστικά από προσωπικό με τεχνική κατάρτιση στην επιστήμη των υπολογιστών (προγραμματιστές, μηχανικοί λογισμικού). Αντικείμενο των τμημάτων αυτών είναι ο αρχιτεκτονικός σχεδιασμός, η ανάπτυξη καθώς και η συντήρηση - επιδιόρθωση σφαλμάτων τόσο του βασικού προϊόντος της εταιρείας όσο και των περιφερειακών υποστηρικτικών προϊόντων.

Σκοπός του τμήματος είναι η ορθή λειτουργία των προϊόντων με τη διαρκή προσθήκη νέων δυνατοτήτων για την κάλυψη των αυξανόμενων απαιτήσεων. Επίσης τα τμήματα αυτά είναι επιφορτισμένα με την παρακολούθηση της εξέλιξης της τεχνολογίας με στόχο την υιοθέτηση νέων πρακτικών και βελτιώσεων της υφιστάμενης τεχνολογικής πλατφόρμας.

- Υποστήριξη πελατών

Αντικείμενο του τμήματος είναι η εξυπηρέτηση - υποστήριξη των πελατών κυρίως απομακρυσμένα. Με κατάλληλα εργαλεία παρέχουν βοήθεια διαφορετικού επιπέδου στον εκάστοτε πελάτη. Είναι το τμήμα που έρχεται πρώτο σε επαφή με τους πελάτες για την καταγραφή των νέων απαιτήσεων, τη συλλογή σφαλμάτων που εντοπίζονται από τους πελάτες και αρκετές φορές γίνονται οι δέκτες παραπόνων λόγω αυτών. Το τμήμα αποτελείται από προσωπικό με γνώση του προϊόντος, της επιχειρησιακής λογικής καθώς και στοιχειώδους τεχνικής κατάρτισης.

- Τμήματα Έργων

Αποτελείται από ομάδες έμπειρων συμβούλων στο χώρο της μηχανογράφησης που αναλαμβάνουν στο σύνολό τους έργα υλοποίησης ενσωμάτωσης του προϊόντος σε νέους πελάτες ή την αναβάθμιση ήδη υφιστάμενων εγκαταστάσεων. Οι ομάδες απαρτίζονται τόσο από επιχειρησιακούς όσο και από τεχνικούς συμβούλους.

Αρμοδιότητα του τμήματος είναι η καταγραφή και ανάλυση των απαιτήσεων ενός νέου πελάτη. Στην συνέχεια γίνεται μια αποσαφήνιση των διαδικασιών του και η αντιστοίχιση με τις αντίστοιχες διαδικασίες που προβλέπονται στο υπάρχον πρόγραμμα. Όπου κρίνεται απαραίτητο οι σύμβουλοι της κάθε ομάδας παραμετροποιούν το πρόγραμμα ώστε να ανταποκρίνεται στις ανάγκες του πελάτη ενώ οι τεχνικοί σύμβουλοι αναπτύσσουν νέες εξειδικευμένες λύσεις προγραμματιστικά αποκλειστικά για τον εκάστοτε πελάτη «*custom solutions*».

### 3.3 Agile προσέγγιση

#### 3.3.1 Εισαγωγή

Ένα ολοκληρωμένο business logic σύστημα επιχειρησιακού σχεδιασμού – μηχανογράφησης «*ERP*» λειτουργεί ως ομπρέλα για την επιχείρηση. Σε οποιονδήποτε κλάδο δραστηριοποίησης και ανεξαρτήτως μεγέθους μεγαλώνει μαζί της και οφείλει να καλύπτει το σύνολο των εμπορικών και οικονομικών δραστηριοτήτων της, από την παραγωγή και τις προμήθειες έως τις πωλήσεις και τη λογιστική καθώς και από την οικονομική και κοστολογική διαχείριση έως τη διαχείριση μισθοδοσίας και προσωπικού.

Αυτομάτως οι απαιτήσεις και οι ανάγκες που καλούνται να καλύψουν αυτά τα πληροφοριακά συστήματα καθιστούν τόσο την ανάπτυξη όσο και την συντήρησή τους μια σύνθετη διαδικασία που απαιτεί καλή οργάνωση, σωστά μελετημένες διαδικασίες όσο και προσοχή καθώς οποιοδήποτε λάθος μπορεί να έχει μεγάλο αρνητικό αντίκτυπο στην εύρυθμη λειτουργία του πελάτη – χρήστη και να τον θέσει αναξιόπιστο απέναντι τόσο στους συναλλασσόμενους του όσο και στις υποχρεώσεις του απέναντι στη πολιτεία.

Στην προσπάθεια να διατηρηθεί το προϊόν σε υψηλό επίπεδο ισάξιο του ανταγωνισμού λύσεων η εταιρεία αποφάσισε την υιοθέτηση της agile μεθοδολογίας για την ανάπτυξη – συντήρησή του. Στόχος ήταν να διατηρηθούν οι παραδοσιακές δυνατότητές του λειτουργικές και ταυτόχρονα να επιτευχθεί η αύξηση της αξίας του με την προσθήκη νέων καινοτόμων.

Στην παρούσα εργασία θα γίνει μελέτη των agile ρόλων καθώς και διαδικασιών που αφορούν το τμήμα έρευνας και ανάπτυξης R&D του ERP συστήματος.



### 3.3.2 Οντότητες

Για την εφαρμογή της agile μεθοδολογίας χρειάστηκε να γίνει επένδυση από την εταιρεία προκειμένου να εκπαιδευτούν οι άμεσα εμπλεκόμενοι εργαζόμενοι. Επιπλέον, δόθηκε το απαραίτητο χρονικό περιθώριο ώστε όλοι να αντιληφθούν τη σημασία εφαρμογής συγκεκριμένων διαδικασιών που απαιτούνται στην agile μεθοδολογία.

Στην προσπάθεια να υιοθετηθεί η agile φιλοσοφία στη «γραμμή παραγωγής» της εταιρείας ήταν απαραίτητο τα στελέχη της να αποκτήσουν αρμοδιότητες σύμφωνα με τις αρχές της μεθοδολογίας με σκοπό την καλύτερη εφαρμογή της. Για να επιτευχθεί αυτό ορισμένοι υπάλληλοι χρειάστηκε να αλλάξουν ελαφρώς αρμοδιότητες ώστε να εναρμονιστούν με τις διαδικασίες που ορίζει η agile φιλοσοφία ενώ άλλοι απέκτησαν επιπλέον υποχρεώσεις σε σχέση με τα καθήκοντα που είχαν έως τότε.

Οι agile ρόλοι που συναντάει κάποιος στην εταιρεία είναι οι εξής:

- **Product Owner:** Το συγκεκριμένο ρόλο ανέλαβε το τμήμα διαχείρισης προϊόντος (product management). Το τμήμα αποτελείται από άτομα με βαθιά γνώση του προϊόντος και του ανταγωνισμού. Στις ήδη αρμοδιότητες που είχαν σχετικά με τη συλλογή των απαιτήσεων για νέες προσθήκες και τις επιδιορθώσεις bugs προστέθηκε η δημιουργία και συντήρηση του product backlog προσθέτοντας όλα τα αποδεκτά αιτήματα για παρέμβαση στο λογισμικό. Το τμήμα αυτό γίνεται πλέον αποδέκτης όλων των εσωτερικών και εξωτερικών αιτημάτων, θέτει τα κριτήρια αποδοχής για τις νέες προσθήκες και αντιπροσωπεύει τις ανάγκες του πελάτη όσον αφορά την παραγωγή.
- **Scrum Master:** Έμπειροι προγραμματιστές τόσο με γνώσεις business όσο και με προγραμματιστικές ικανότητες επωμίστηκαν τις αρμοδιότητες του scrum master παράλληλα με τον ρόλο του team leader. Στα βασικά καθήκοντά τους είναι η εύρυθμη λειτουργία της ομάδας την οποία επιβλέπουν. Δίνουν λύσεις τόσο σε τεχνικό επίπεδο προγραμματιστικά, σε πιθανά προβλήματα εξοπλισμού όταν προκύπτουν και εμποδίζουν την παραγωγική διαδικασία ενώ παράλληλα εξασφαλίζουν ότι η ομάδα εφαρμόζει όσο το δυνατόν καλύτερα τις διαδικασίες του agile.

- **Scrum Team:** Το τμήμα έρευνας και ανάπτυξης χωρίστηκε σε ομάδες ανάπτυξης υπό την επίβλεψη των team leaders. Κάθε τμήμα απαρτίζεται από τέσσερα έως πέντε άτομα. Έγινε προσπάθεια ώστε τα άτομα κάθε ομάδας να έχουν επικαλυπτόμενες γνώσεις και να υπάρχουν όλα τα επίπεδα εμπειρίας προκειμένου να μοιράζεται η τεχνογνωσία εύκολα και να καθιστά την ομάδα αυτόνομη.

### 3.3.3 Διαδικασίες

Δεδομένου ότι η επιχείρηση λειτουργούσε για πολλά χρόνια με δικό της μοντέλο ανάπτυξης χρειάστηκε χρόνος και προσπάθεια από όλα τα εμπλεκόμενα μέρη ώστε να προσαρμόσουν το agile μοντέλο στο δικό της τρόπο εργασίας. Όπως είναι λογικό δεν ήταν εύκολο να εφαρμοστεί κατά γράμμα το σύνολο των διαδικασιών που ορίζει το agile και αυτό οφείλεται εκτός από τις εσωτερικές διαδικασίες και στην πολιτική που ακολουθούσε η εταιρεία με το εξωτερικό της περιβάλλον. Ήταν δύσκολο να προσαρμοστούν οι συνεργάτες και οι πελάτες στον νέο τρόπο ανάπτυξης και αντιμετώπισης των απαιτήσεών τους. Ωστόσο το μεγάλο πλεονέκτημα του agile είναι ότι βασίζεται στην ευελιξία, επομένως με τον καιρό υπήρξε ισορροπία και ικανοποίηση των απαιτήσεων τόσο μεγάλων όσο και μικρότερων πελατών.

Ορίστηκαν ως χρόνος επανάληψης οι δύο βδομάδες και σύμφωνα με τη διάρκεια αυτή γίνονται οι εκτιμήσεις για τη προσπάθεια που απαιτείται προκειμένου να γίνει η ανάπτυξη των νέων χαρακτηριστικών ενώ μέσα στο χρονικό διάστημα αυτό αφιερώνεται χρόνος - προσπάθεια για την επίλυση προβλημάτων των υπάρχοντων χαρακτηριστικών του προϊόντος.

Οι agile διαδικασίες που εφαρμόζονται στην εταιρεία είναι οι εξής:

- **Συνεδρίαση για το σχεδιασμό της επανάληψης (sprint planning meeting):** Η συγκεκριμένη διαδικασία έχει διπλό ρόλο. Στα πλαίσια σχεδιασμού της επανάληψης πραγματοποιείται μια συνάντηση κατά τη διάρκεια ενός sprint μεταξύ των scrum masters και του product owner. Στη συνάντηση αυτή συζητούνται τα θέματα που θα ανατεθούν στις ομάδες για το επόμενο sprint σύμφωνα με την προτεραιότητα που έχουν λάβει στο backlog και στην ουσία μεταφέρονται από το product backlog στο επόμενο sprint backlog. Στα πλαίσια του σχεδιασμού της

επανάληψης γίνεται παράλληλα και μια πρώτη **εκτίμηση της προσπάθειας** που θα χρειαστούν τα μέλη κάθε ομάδας προκειμένου να φέρουν εις πέρας τα αιτήματα που τους ανατίθενται.

- **Καθημερινή συνεδρίαση ομάδας (daily meeting):** Από την αρχή της προσπάθειας εφαρμογής του agile μοντέλου θεσπίστηκε το καθημερινό πρωινό «*stand up meeting*» κατά τη διάρκεια του οποίου τα μέλη της ομάδας αναφέρουν τι έκαναν την προηγούμενη ημέρα, που συνάντησαν δυσκολία και ενημερώνουν με τι θα απασχοληθούν κατά τη διάρκεια της ημέρας. Εκτός ελάχιστων εξαιρέσεων μη πραγματοποίησης από κάποια ομάδα, η καθημερινή συνεδρίαση των ομάδων αποτελεί βασικό συστατικό του agile που είναι αναπόσπαστο μέρος των διαδικασιών του τμήματος έρευνας και ανάπτυξης.
- **Συνεδρίαση στο τέλος της επανάληψης (sprint review):** Αποτελεί βασικό πυλώνα στις διαδικασίες του agile και μια από τις σημαντικότερες διαδικασίες που εφαρμόζονται στην εταιρεία που ξεπερνάει τα όρια του τμήματος ανάπτυξης. Πραγματοποιείται κάθε δύο εβδομάδες και διαδραματίζει το μέσο επικοινωνίας μεταξύ του τμήματος ανάπτυξης με όλα τα εμπλεκόμενα τμήματα. Κατά τη διάρκεια της συνάντησης ενημερώνονται άμεσα όλοι σχετικά με απαιτήσεις που είχαν ζητηθεί και αναπτύχθηκαν στη διάρκεια του sprint. Δίνονται διευκρινίσεις για απαιτήσεις που είναι ήδη σε εξέλιξη ανάπτυξης ενώ λαμβάνεται και ένα πρώτο feedback.

Στην μέχρι τώρα εφαρμογή της μεθόδου agile διαδικασίες όπως αυτή της retrospective συνάντησης και της βαθμολόγησης της προσπάθειας των απαιτήσεων από την ίδια την ομάδα δεν πραγματοποιούνται σε σταθερή βάση. Παροδικά γίνεται προσπάθεια εφαρμογής των πρακτικών αυτών σε αντίθεση με τις υπόλοιπες διαδικασίες που έχουν παγιωθεί και πλέον αποτελούν αναπόσπαστο κομμάτι κατά την παραγωγική διαδικασία. Όσο ο βαθμός ωρίμανσης της υιοθέτησης της agile φιλοσοφίας αυξάνεται τόσο και οι υπόλοιπες πρακτικές της μεθόδου θα παγιώνονται στην συνολική διαδικασία ανάπτυξης και συντήρησης του προϊόντος.

## Κεφάλαιο 4<sup>ο</sup>

Στο κεφάλαιο αυτό γίνεται μια παρουσίαση των δεικτών απόδοσης (*Key Performance Indicators*) σχετικά με το τμήμα έρευνας & ανάπτυξης. Για την δημιουργία των δεικτών KPIs έγινε συλλογή και ανάλυση δεδομένων από τη πλατφόρμα καταγραφής και διαχείρισης των απαιτήσεων του εξωτερικού περιβάλλοντος (συνεργάτες - τελικοί πελάτες).

RefId	Status	Priority	UpdateDate	Ok Date	Ok Year	YouTrack ID	SpecManager	R_D Team	Τιτλή/Περιγραφή
1	Accepted	Not Scheduled	07/05/2020 10:40		0	ERP-10568	Κουτσουπας Αθανάσιος	0,234,232	Invalid Date
2	Accepted	Not Scheduled	07/05/2020 10:40		0	ERP-7232	Κουτσουπας Αθανάσιος	401,232	Invalid Date
3	Accepted	Not Scheduled	04/05/2020 06:51		0	ERP-5644	Κουτσουπας Αθανάσιος	221,111,232,171	New Spec
4	Accepted	Not Scheduled	22/01/2021 09:39		0	ERP-10458	Κουτσουπας Αθανάσιος	0,302,232	New Spec
5	Accepted	Not Scheduled	04/05/2017 11:54		0		Κουτσουπας Αθανάσιος	232	New Spec
6	Accepted	Not Scheduled	28/12/2017 00:00		0		Κουτσουπας Αθανάσιος	232	New Spec
7	Accepted	Not Scheduled	03/06/2016 13:25		0		Κουτσουπας Αθανάσιος	196,232	New Spec
8	Accepted	Not Scheduled	19/04/2017 11:23		0		Κουτσουπας Αθανάσιος	232	New Spec
9	Accepted	Not Scheduled	29/01/2018 00:00		0	ERP-399	Κουτσουπας Αθανάσιος	232	New Spec
10	Accepted	Not Scheduled	18/10/2018 00:00		0	ERP-2375	Κουτσουπας Αθανάσιος	232	New Spec
11	Accepted	Not Scheduled	08/03/2019 09:27		0		Κουτσουπας Αθανάσιος	315,232	New Spec
12	Accepted	Not Scheduled	02/07/2015 00:00		0		Κουτσουπας Αθανάσιος	240,232	New Spec
13	Accepted	Not Scheduled	09/08/2019 08:00		0		Κουτσουπας Αθανάσιος	232	New Spec
14	Accepted	Not Scheduled	16/05/2018 11:35		0		Κουτσουπας Αθανάσιος	160,232	New Spec
15	Accepted	Not Scheduled	14/10/2018 00:00		0		Κουτσουπας Αθανάσιος	232	New Spec
16	Accepted	Not Scheduled	19/07/2017 00:00		0		Κουτσουπας Αθανάσιος	232	New Spec
17	Accepted	Not Scheduled	11/09/2019 09:47		0		Κουτσουπας Αθανάσιος	232	New Spec
18	Accepted	Not Scheduled	18/10/2018 00:00		0		Κουτσουπας Αθανάσιος	160,232	New Spec
19	Accepted	Not Scheduled	18/10/2018 00:00		0		Κουτσουπας Αθανάσιος	232	New Spec
20	Accepted	Not Scheduled	14/01/2020 06:40		0		Κουτσουπας Αθανάσιος	111,232	New Spec
21	Accepted	Not Scheduled	19/06/2020 00:00		0	ERP-1231	Κουτσουπας Αθανάσιος	232	New Spec
22	Accepted	Not Scheduled	02/05/2019 13:34		0	ERP-1842	Κουτσουπας Αθανάσιος	232	New Spec
23	Accepted	Not Scheduled	18/09/2019 11:09		0		Κουτσουπας Αθανάσιος	232	New Spec
24	Accepted	Not Scheduled	02/06/2020 16:40		0		Κουτσουπας Αθανάσιος	232	New Spec
25	Accepted	Not Scheduled	11/12/2017 09:58		0		Κουτσουπας Αθανάσιος	193,232	New Spec
26	Accepted	Not Scheduled	22/03/2021 14:47		0	ERP-9680	Κουτσουπας Αθανάσιος	0,112,232	New Spec
27	Accepted	Not Scheduled	13/05/2021 13:53		0	ERP-11266	Κουτσουπας Αθανάσιος	0,112,232	New Spec
28	Accepted	Not Scheduled	06/05/2021 13:48		0	ERP-11544	Κουτσουπας Αθανάσιος	232	New Spec
29	Accepted	Not Scheduled	13/05/2021 13:51		0	ERP-10379	Κουτσουπας Αθανάσιος	0,112,232	New Spec
30	Accepted	Not Scheduled	02/12/2015 00:00	01/12/2015 00:00	2015		Κουτσουπας Αθανάσιος	196,232	New Spec

Εικόνα 10: Softone ticketing system

### 4.1 Συλλογή Δεδομένων Εξωτερικού Περιβάλλοντος

Τα δεδομένα που συλλέχθηκαν για τις ανάγκες δημιουργίας των δεικτών αφορούν τα τελευταία έξι χρόνια λειτουργίας της εταιρείας (2015-2020) και είναι τα εξής:

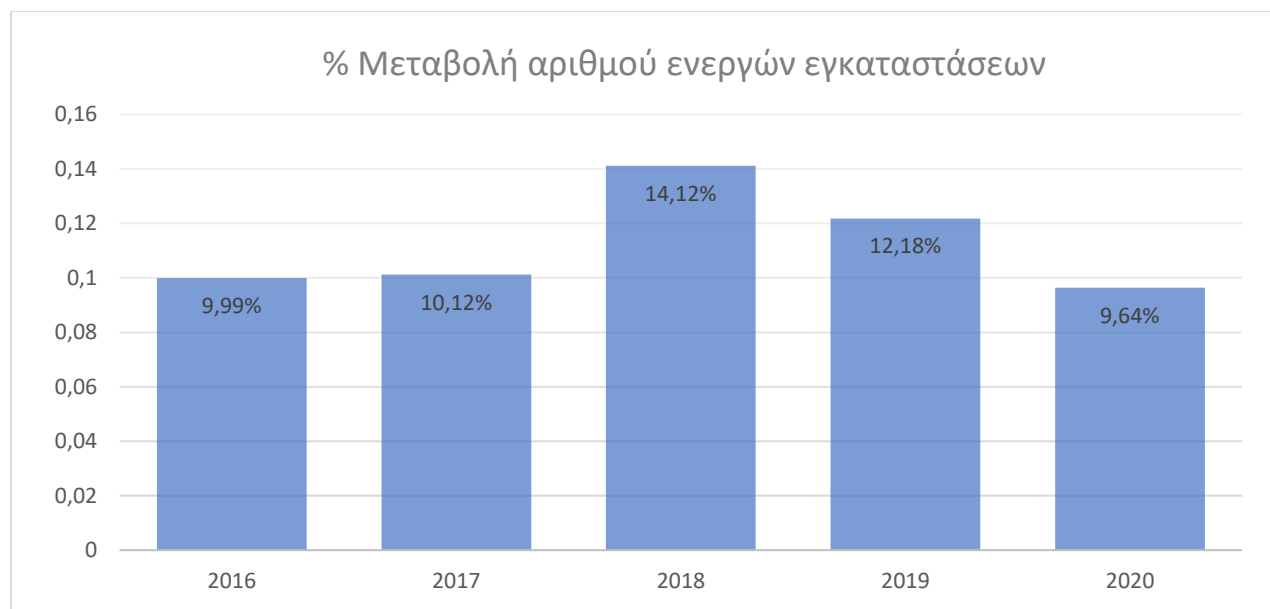
- Ο μέσος όρος των εργαζομένων στο τμήμα R&D Core της εταιρείας ανά έτος.
- Ο μέσος όρος γραμμών κώδικα του λογισμικού Softone ERP ανά έτος.
- Ο συνολικός αριθμός απαιτήσεων νέων προδιαγραφών (*new specs*) που καταγράφηκε στην πλατφόρμα της εταιρείας είτε από το εξωτερικό περιβάλλον είτε από το εσωτερικό, είτε στα πλαίσια γενικής αναβάθμισης του προϊόντος ανά έτος.
- Ο αριθμός των νέων προδιαγραφών που υλοποιήθηκαν ανά έτος.
- Ο συνολικός αριθμός δυσλειτουργιών που καταγράφηκαν από τους πελάτες κατά τη διάρκεια της διαδικασίας ανάπτυξης ανά έτος.

- Ο συνολικός αριθμός σφαλμάτων - δυσλειτουργιών που επιλύθηκαν ανά έτος.
- Ο αριθμός ενεργών εγκαταστάσεων στο τέλος κάθε έτους.

Στον παρακάτω πίνακα αποτυπώνονται ορισμένα από τα δεδομένα όπως συλλέχθηκαν προκειμένου να υπολογιστούν σύμφωνα με αυτά οι δείκτες απόδοσης.

	2015	2016	2017	2018	2019	2020
Προγραμματιστές	9	9	11	11	13	16
Απαιτήσεις νέων προδιαγραφών	510	558	607	688	759	829
Υλοποιημένες νέες προδιαγραφές	285	321	359	439	511	590
Γραμμές κώδικα	3.361.999	3.459.714	3.631.629	3.783.891	4.176.202	4.382.614

Πίνακας 1: Δεδομένα εξωτερικού περιβάλλοντος



Διάγραμμα 1: Ποσοστιαία μεταβολή ενεργών εγκαταστάσεων

Στο σημείο αυτό πρέπει να αναφερθεί ότι το τμήμα R&D άρχισε να εφαρμόζει την μέθοδο agile στις αρχές του 2018. Τα προηγούμενα χρόνια δεν υπήρχε κάποιος αυστηρά καθορισμένος τρόπος λειτουργίας.

#### 4.2 Ποσοτικοί Δείκτες απόδοσης (KPIs)

Η δημιουργία δεικτών που σχετίζονται με το τμήμα έρευνας και ανάπτυξης της εταιρείας είναι ενδιαφέρουσα καθώς μέσα από τη μελέτη και τη σύγκριση των αποτελεσμάτων τους μπορεί να προκύψει το συμπέρασμα κατά πόσο η agile μεθοδολογία αποτελεί καλή και αποδοτική πρακτική για την εταιρεία στο πέρασμα του χρόνου καθώς επίσης και για το κατά πόσο συνέβαλε στην βελτίωση της ανάπτυξης του προϊόντος.

Η εξέλιξη της κατάστασης των αιτήσεων που έχουν υποβληθεί και έχουν επιβεβαιωθεί είναι προσβάσιμη από τους πελάτες, οι οποίοι ενημερώνονται σε εβδομαδιαία βάση για το ποια από τις αιτήσεις τους έχουν εξυπηρετηθεί μέσω μιας διαδικτυακής διεπαφής (portal).

##### 4.2.1 Αριθμός νέων προδιαγραφών ανά εγκατάσταση

###### 4.2.1.1 Ανάγκη

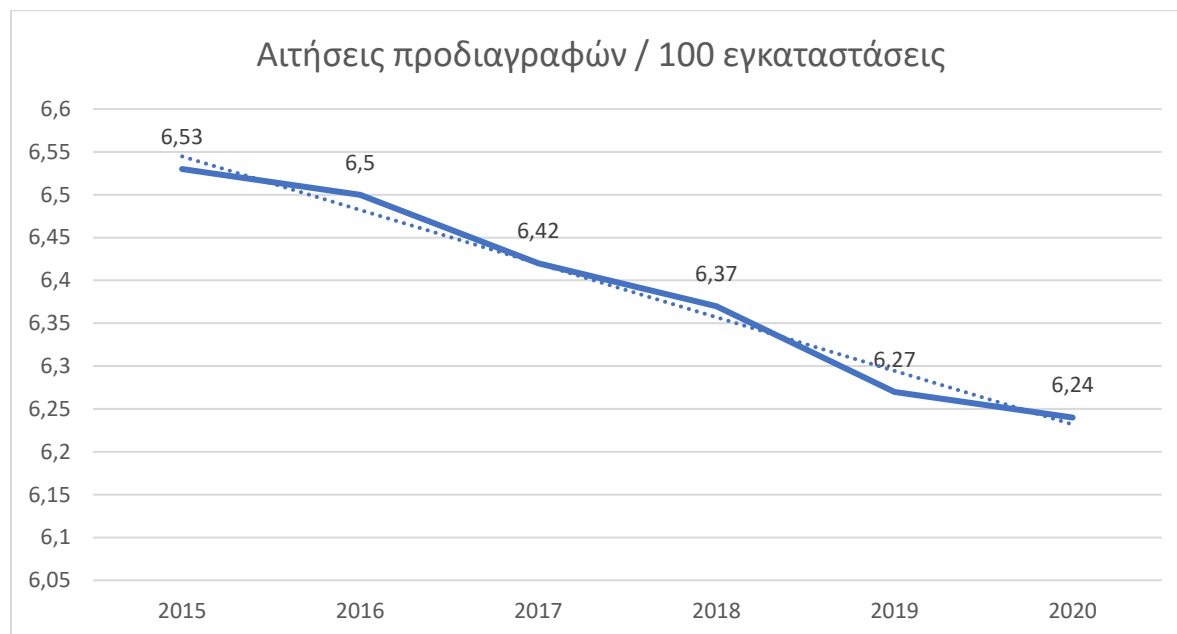
Καθώς η εταιρεία αναπτύσσεται και το πρόγραμμα αποκτά μια φήμη αξιοπιστίας και αποτελεσματικότητας κερδίζει μεγαλύτερο μερίδιο της αγοράς. Αυτό έχει σαν αποτέλεσμα να μεγαλώνει διαρκώς το φάσμα των εταιρειών που προτίθενται να το χρησιμοποιήσουν. Προκειμένου να εγκατασταθεί και να λειτουργήσει ευεργετικά σε εταιρείες διαφορετικών επιχειρηματικών δραστηριοτήτων είναι απαραίτητη η ανάπτυξη νέων προδιαγραφών.

Στον αντίποδα, όσο το πρόγραμμα ωριμάζει οι δυνατότητές του αυξάνονται διαρκώς καθιστώντας το πλήρες και επαρκές για να ανταποκριθεί στο μεγαλύτερο μέρος της αγοράς και να στέκεται επάξια δίπλα στον ανταγωνισμό.

Ορίζοντας έναν δείκτη που αφορά τα αιτήματα για ανάπτυξη και προσθήκη νέων προδιαγραφών, μπορεί να γίνει καλύτερος έλεγχος για το πόσο το πρόγραμμα θεωρείται ικανό να καλύψει το διαρκώς αυξανόμενο πελατολόγιο και παράλληλα δίνεται η αίσθηση για το τι δυναμική υπάρχει στην είσοδο νέων επιχειρηματικών κλάδων.

#### 4.2.1.2 Ανάλυση

Ο συγκεκριμένος δείκτης υπολογίζεται από το κλάσμα των καταγεγραμμένων αιτημάτων για νέες προσθήκες ως προς το πλήθος των ενεργών εγκαταστάσεων στο τέλος του κάθε έτους. Για ευκολότερη επεξήγηση γίνεται με αναγωγή στην εκατοντάδα όπως φαίνεται στο παρακάτω διάγραμμα.



Διάγραμμα 2: Αιτήσεις για προσθήκη νέων προδιαγραφών

#### 4.2.1.3 Συμπεράσματα

Όπως γίνεται αντιληπτό κατά τη διάρκεια των ετών και όσο το πρόγραμμα ωριμάζει μειώνεται ο αριθμός των αιτημάτων για νέες προδιαγραφές. Παρά το γεγονός ότι η αύξηση των νέων εγκαταστάσεων από το 2015 στο 2020 υπολογίζεται στο 70% σύμφωνα με το *διάγραμμα 1*, το ότι μειώνονται έστω και με μικρό ρυθμό οι αιτήσεις δείχνει ότι το πρόγραμμα μπορεί να ανταπεξέλθει ικανοποιητικά στους επαγγελματικούς τομείς που δραστηριοποιούνται οι πελάτες που το χρησιμοποιούν.

Στο σύνολο των αιτημάτων δεν είναι εύκολο να υπολογιστούν υποχρεωτικές αλλαγές / προσθήκες που αφορούν σε νομοθετικές αλλαγές είτε σε θέματα εμπορικής διαχείρισης (πχ τέλος πλαστικής σακούλας) είτε σε αλλαγές σε θέματα μισθοδοσίας.

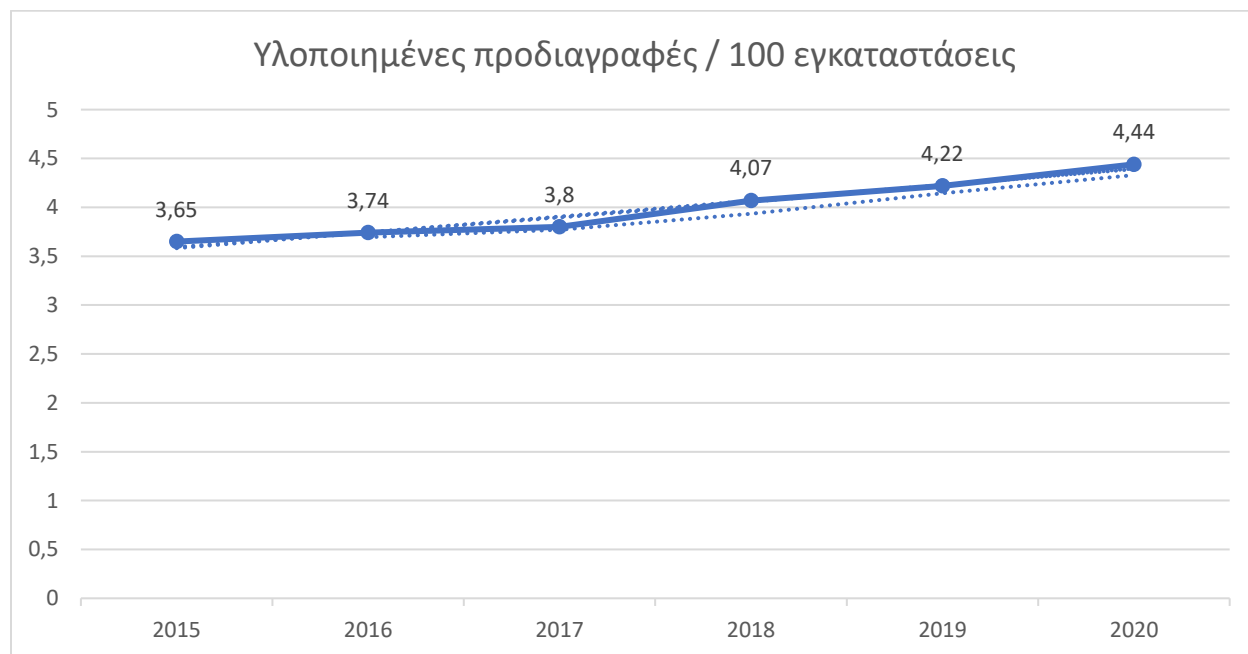
## 4.2.2 Αριθμός υλοποιημένων προδιαγραφών ανά εγκατάσταση

### 4.2.2.1 Ανάγκη

Αντίστοιχο ενδιαφέρον έχει η μελέτη του αριθμού των νέων προδιαγραφών που υλοποιήθηκαν από το τμήμα έρευνας και ανάπτυξης. Ορίζοντας έναν δείκτη που αφορά τα αιτήματα που εξυπηρετήθηκαν με προσθήκες προδιαγραφών, μπορεί να γίνει καλύτερος έλεγχος για την συνολική ικανοποίηση των τελικών πελατών καθώς επίσης να οριστεί από τη διοίκηση ο ελάχιστος απαιτούμενος αριθμός υλοποιήσεων προκειμένου να καλύπτονται οι στόχοι που έχουν τεθεί από τμήματα πωλήσεων.

### 4.2.2.2 Ανάλυση

Ο συγκεκριμένος δείκτης υπολογίζεται από το κλάσμα του πλήθους των υλοποιημένων αιτημάτων νέων προδιαγραφών τα οποία έχουν επιλεγθεί και έχουν γίνει αποδεκτά από τους πελάτες, ως προς το πλήθος των ενεργών εγκαταστάσεων στο τέλος του κάθε έτους. Για ευκολότερη επεξήγηση γίνεται με αναγωγή στην εκατοντάδα όπως φαίνεται στο παρακάτω διάγραμμα.



Διάγραμμα 3: Υλοποίηση νέων προδιαγραφών



#### 4.2.2.3 Συμπεράσματα

Όπως είναι αναμενόμενο η αύξηση της αποδοτικότητας είναι εμφανής. Επίσης παρατηρείται μια μικρή διαφοροποίηση όσον αφορά την περίοδο πριν και μετά την εφαρμογή της agile μεθοδολογίας. Σε συνδυασμό με την μείωση απαιτήσεων για νέες προδιαγραφές το πρόγραμμα στα επόμενα χρόνια θα θεωρείται απολύτως ώριμο όσον αφορά τις άμεσες ανάγκες των πελατών.

### 4.2.3 Αριθμός καταγεγραμμένων δυσλειτουργιών ανά εγκατάσταση

#### 4.2.3.1 Ανάγκη

Βασικό κριτήριο για την ικανοποίηση των χρηστών είναι η αξιοπιστία του προγράμματος πέρα από το σύνολο των δυνατοτήτων που προσφέρει. Απαραίτητη προϋπόθεση για να επιτευχθεί αυτό είναι ο συστηματικός έλεγχος των νέων προδιαγραφών που ενσωματώνονται στην υφιστάμενη υλοποίηση προκειμένου να μην δημιουργηθούν δυσλειτουργίες - σφάλματα τα οποία θα δημιουργήσουν προβλήματα στην απρόσκοπτη λειτουργία του προγράμματος από τη πλευρά του τελικού χρήστη.

Τα σφάλματα – δυσλειτουργίες αφού καταγραφούν και επιβεβαιωθούν από το αρμόδιο τμήμα κατηγοριοποιούνται. Βασικά κριτήρια κατηγοριοποίησης είναι τόσο ο αριθμός όσο και η σπουδαιότητα των πελατών τους οποίους επηρεάζουν. Επίσης εξετάζεται αν υπάρχει τρόπος έστω και προσωρινής προσπέρασής τους.

Ορίζοντας έναν δείκτη που αφορά τα καταγεγραμμένα και επιβεβαιωμένα σφάλματα που έχουν αναφερθεί από το εξωτερικό περιβάλλον της εταιρείας μπορεί να γίνει καλύτερος έλεγχος για το πόσο αξιόπιστο θεωρείται το πρόγραμμα κάνοντας παρεμβάσεις από τους αρμόδιους ή θεσπίζοντας διαδικασίες προκειμένου να εξασφαλιστεί ή ποιότητά του.

#### 4.2.3.2 Ανάλυση

Ο συγκεκριμένος δείκτης υπολογίζεται από το κλάσμα των καταγεγραμμένων δυσλειτουργιών ως προς το πλήθος των ενεργών εγκαταστάσεων στο τέλος του κάθε έτους. Για ευκολότερη επεξήγηση γίνεται με αναγωγή στην εκατοντάδα όπως φαίνεται στο παρακάτω διάγραμμα.



Διάγραμμα 4: Δυσλειτουργίες ανά 100 εγκαταστάσεις

#### 4.2.3.3 Συμπεράσματα

Όπως φαίνεται στα έτη 2015-2017 παράλληλα με την ανάπτυξη του προγράμματος και την προσθήκη νέων προδιαγραφών εισάγονταν στο πρόγραμμα και δυσλειτουργίες. Ωστόσο από την πρώτη κιόλας χρονιά εφαρμογής της μεθόδου agile παρατηρείται αισθητή μείωση των σφαλμάτων και ειδικά στο έτος 2020 όπου οι εγκαταστάσεις είναι κατά 70% περισσότερες από ότι το 2015 τα καταγεγραμμένα σφάλματα είναι λιγότερα ανά εγκατάσταση από ότι το πρώτο έτος της μελέτης. Αυτό αποδεικνύει ότι θέτοντας κριτήρια αποδοχής για κάθε νέα υλοποίηση από τη πλευρά των product owners μειώθηκαν οι δυσλειτουργίες σε σχέση με τα προηγούμενα έτη και όπως φαίνεται από την τάση τα σφάλματα θα περιοριστούν περαιτέρω στο μέλλον.

#### 4.2.4 Δυσλειτουργίες σε σχέση με το μέγεθος του κώδικα

##### 4.2.4.1 Ανάγκη

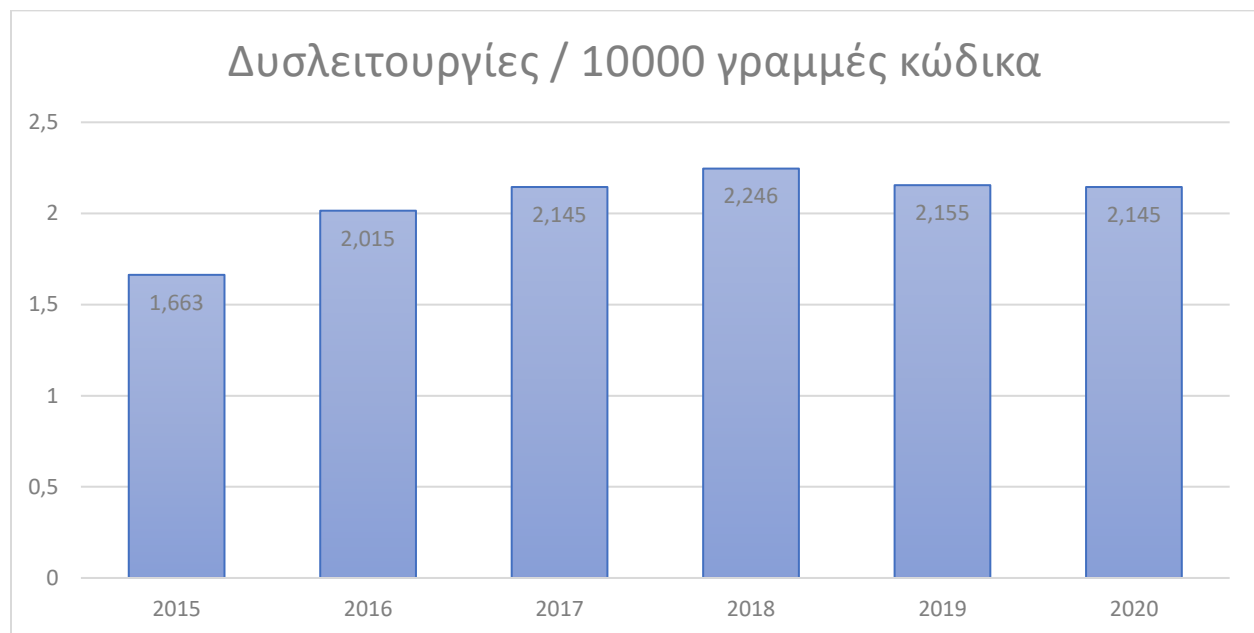
Ένας άλλος δείκτης που έχει αξία να μελετηθεί αφορά τα καταγεγραμμένα σφάλματα σε σχέση με το μέγεθος του κώδικα. Δυσλειτουργίες προκύπτουν διαρκώς κατά τη διαδικασία ανάπτυξης. Εσφαλμένες συμπεριφορές του προγράμματος μπορεί να προκύψουν για διάφορους λόγους. Η ελλιπής τεκμηρίωση μίας νέας απαίτησης ή η λανθασμένη αντίληψη ενός προβλήματος από τους

προγραμματιστές του τμήματος έρευνας και ανάπτυξης μπορεί εύκολα να προκαλέσει μια λανθασμένη συμπεριφορά του προγράμματος. Στα πολύπλοκα συστήματα όπως είναι τα ERPs συχνά παρατηρείται το φαινόμενο η επίλυση ενός σφάλματος που έχει αναφερθεί από έναν πελάτη να δημιουργεί λανθασμένη συμπεριφορά για έναν άλλο.

Ορίζοντας έναν δείκτη που αφορά το πλήθος των σφαλμάτων σε σχέση με το μέγεθος του προγράμματος (γραμμές κώδικα) γίνεται ευκολότερη η παρακολούθηση και η πρόβλεψη πιθανών λαθών. Κατά συνέπεια μπορούν να δημιουργηθούν διαδικασίες ελέγχου και διατήρησης των σφαλμάτων εντός επιθυμητών ορίων.

#### 4.2.4.2 Ανάλυση

Ο συγκεκριμένος δείκτης υπολογίζεται από το κλάσμα των επιβεβαιωμένων σφαλμάτων ως προς το πλήθος του συνολικού αριθμού γραμμών κώδικα. Για τις ανάγκες υπολογισμού χρησιμοποιούμε τα μεγέθη στο τέλος κάθε έτους.



Διάγραμμα 5: Δυσλειτουργίες ανά 1000 γραμμές κώδικα

#### 4.2.4.3 Συμπέρασμα

Αντίστοιχα, στο *διάγραμμα 5* βλέπουμε ότι κατά τη διάρκεια των 3 πρώτων ετών το ποσοστό σφαλμάτων ανά δέκα χίλιες γραμμές κώδικα είναι της τάξεως των 2 σφαλμάτων στις 10000

γραμμές κώδικα με ελαφριά τάση ανόδου. Στα τέλη του 2018 και ενώ εφαρμόζεται πλέον η μέθοδος agile το ποσοστό αυτό βρίσκεται οριακά πάνω από τα 2 σφάλματα ωστόσο παρατηρείται τάση μείωσης. Η αύξηση μεγέθους του προγράμματος και η δημιουργία νέας διεπαφής χρήστη (UI) το 2018 δικαιολογεί την οριακή αύξηση των δυσλειτουργιών καθώς χρειάστηκε ένα εύλογο χρονικό διάστημα μέχρι να σταθεροποιηθεί η απόδοση του προγράμματος.

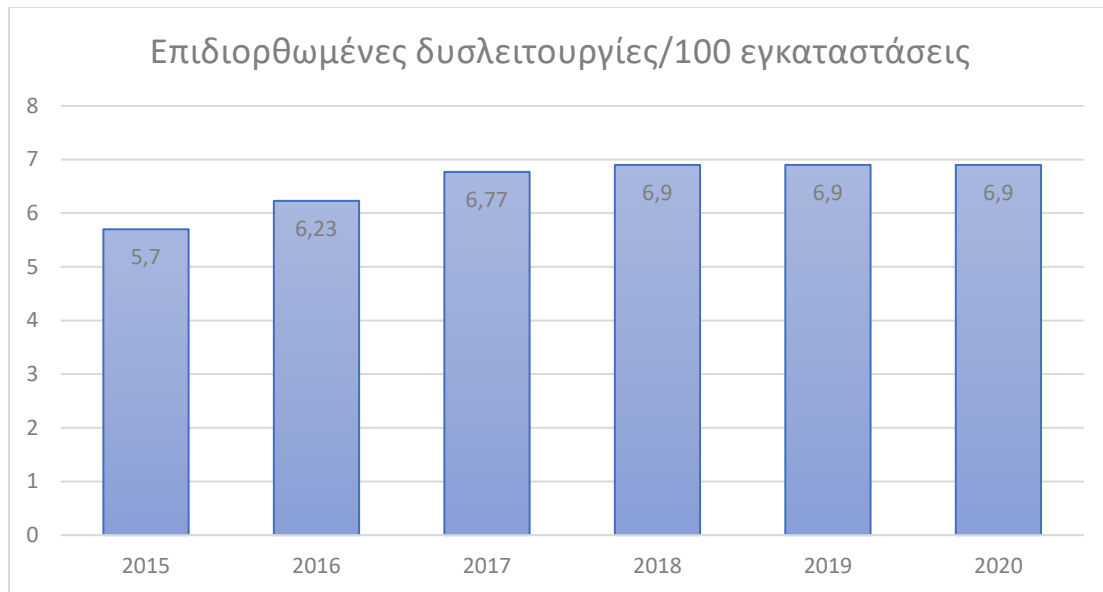
#### 4.2.5 Αριθμός επιλυμένων δυσλειτουργιών ανά εγκατάσταση

##### 4.2.5.1 Ανάγκη

Αντίστοιχο ενδιαφέρον έχει η μελέτη του αριθμού των δυσλειτουργιών που επιλύθηκαν από το τμήμα έρευνας και ανάπτυξης. Ορίζοντας έναν δείκτη που αφορά τα σφάλματα που επιλύθηκαν, μπορεί να γίνει καλύτερος έλεγχος ειδικά για σφάλματα που χαρακτηρίστηκαν επείγοντα και μπορεί να είχαν σημαντική επίπτωση στην λειτουργία της εκάστοτε επιχείρησης. Σύμφωνα με τα αποτελέσματα που θα προκύψουν θα μπορούσε η διοίκηση της εταιρείας να θέσει ως στόχο ένα ελάχιστο νούμερο ικανοποίησης που θα αποτελεί και στοιχείο παρεχόμενης υπηρεσίας με το ανάλογο κόστος.

##### 4.2.5.2 Ανάλυση

Ο συγκεκριμένος δείκτης υπολογίζεται από το κλάσμα του πλήθους των επιδιορθωμένων σφαλμάτων - δυσλειτουργιών που έχουν επιβεβαιωθεί, ως προς το πλήθος των ενεργών εγκαταστάσεων στο τέλος του κάθε έτους. Για ευκολότερη επεξήγηση γίνεται με αναγωγή στην εκατοντάδα όπως φαίνεται στο παρακάτω διάγραμμα.



Διάγραμμα 6: Επιδιορθωμένες δυσλειτουργίες ανά 100 εγκαταστάσεις

#### 4.2.5.3 Συμπεράσματα

Όπως γίνεται αντιληπτό με το πέρασμα του χρόνου ο αριθμός των επιδιορθωμένων σφαλμάτων σταθεροποιείται και αυτό οφείλεται σε πολλούς παράγοντες. Αφενός καθοριστικό ρόλο παίζει το πλήθος των προγραμματιστών και η εμπειρία τους, αφετέρου η σωστά τεκμηριωμένη υλοποίηση των νέων προδιαγραφών με τον απαραίτητο έλεγχο μειώνει τον αριθμό σφαλμάτων όπως είδαμε σε προηγούμενο δείκτη με αποτέλεσμα να κρίνεται ικανοποιητικός ο συγκεκριμένος αριθμός επιδιορθωμένων δυσλειτουργιών

#### 4.2.6 Ανάπτυξη - Αποσφαλμάτωση ανά προγραμματιστή

##### 4.2.6.1 Ανάγκη

Το ERP της εταιρείας έχει πετύχει τα τελευταία χρόνια να κερδίσει μεγάλο μερίδιο της αγοράς. Οι δυνατότητες που παρέχει καλύπτουν μεγάλο εύρος των επιχειρησιακών αναγκών και αυτό έχει σαν συνέπεια να αυξάνεται διαρκώς το πλήθος των πελατών που το εμπιστεύεται.

Το αποτέλεσμα είναι οι απαιτήσεις για βελτιώσεις και προσθήκες να αυξάνονται διαρκώς. Όσο αυξάνεται ο φόρτος εργασίας αυξάνεται και ο χρόνος παράδοσης των απαιτούμενων προσθηκών αλλά και η πιθανότητα για δημιουργία δυσλειτουργιών. Στα πλαίσια της διαρκούς ανάπτυξης της

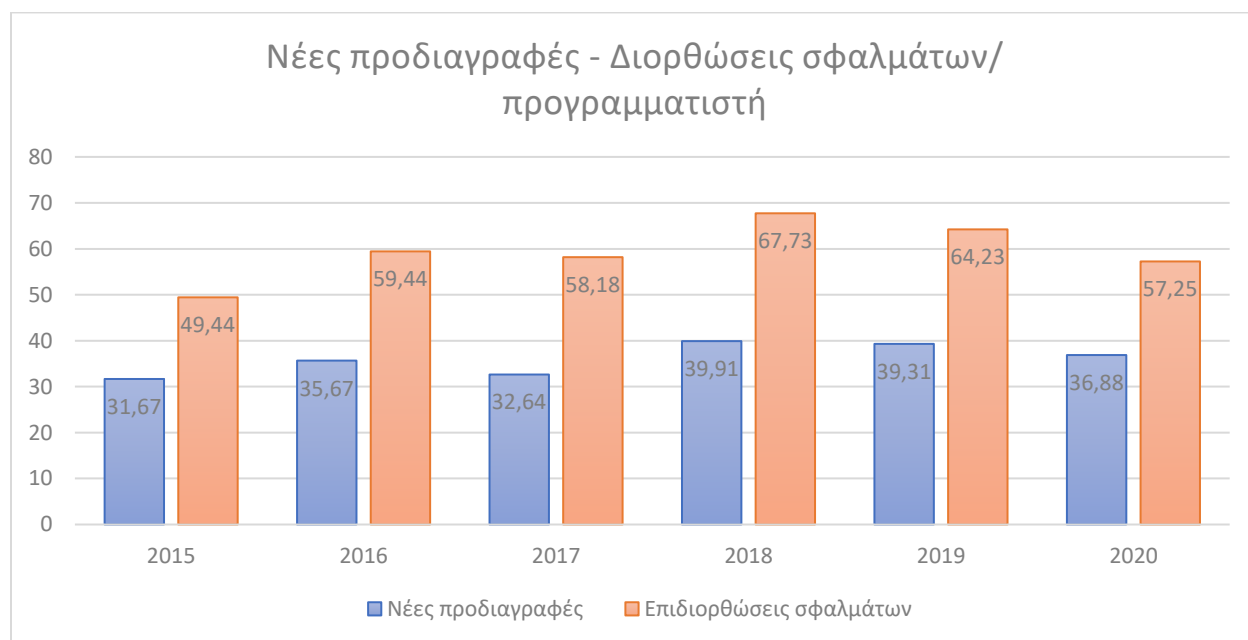
εταιρείας έχει αυξηθεί και το προσωπικό της εταιρείας και κατά συνέπεια και το πλήθος των προγραμματιστών στο τμήμα έρευνας & ανάπτυξης (R&D).

Ορίζοντας δείκτες που αφορούν το πλήθος των νέων προδιαγραφών που μπορεί να ολοκληρώσει κατά μέσο όρο ένας προγραμματιστής καθώς επίσης και των δυσλειτουργιών που λύνει κατά τη διάρκεια του έτους μπορεί να γίνει μία εκτίμηση σχετικά με τη δυναμική του τμήματος.

Σύμφωνα με τον υπολογισμό της «ικανότητας» του τμήματος μπορεί να γίνει σχεδιασμός για την μελλοντική ανάπτυξη του προϊόντος. Μπορεί δηλαδή η εταιρεία να δεσμευτεί σε συνεργάτες και πελάτες για το ποιες προσθήκες μπορεί το τμήμα να φέρει εις πέρας και να γίνει και μια εκτίμηση χρόνου.

#### 4.2.6.2 Ανάλυση

Οι συγκεκριμένοι δείκτες υπολογίζονται από το κλάσμα του συνόλου των υλοποιημένων νέων προδιαγραφών στο τέλος κάθε έτους και των επιδιορθωμένων σφαλμάτων αντίστοιχα με το μέσο αριθμό προγραμματιστών κατά τη διάρκεια του έτους.



Διάγραμμα 7: Νέες προδιαγραφές - Διορθώσεις σφαλμάτων/ προγραμματιστή

#### 4.2.6.3 Συμπεράσματα

Στο *διάγραμμα 6* βλέπουμε ότι κατά το πέρασμα των ετών ο μέσος όρος των νέων προδιαγραφών που ολοκληρώνει κάθε προγραμματιστής έχει αυξητική τάση, ωστόσο θα πρέπει να επισημανθούν ορισμένα σημεία. Κάθε φορά που ο αριθμός προγραμματιστών αυξάνεται, η αύξηση των νέων προδιαγραφών δεν είναι η ανάλογη. Αυτό οφείλεται στο γεγονός ότι είναι απαραίτητος αρκετός χρόνος για την εκπαίδευση ενός νέου στελέχους.

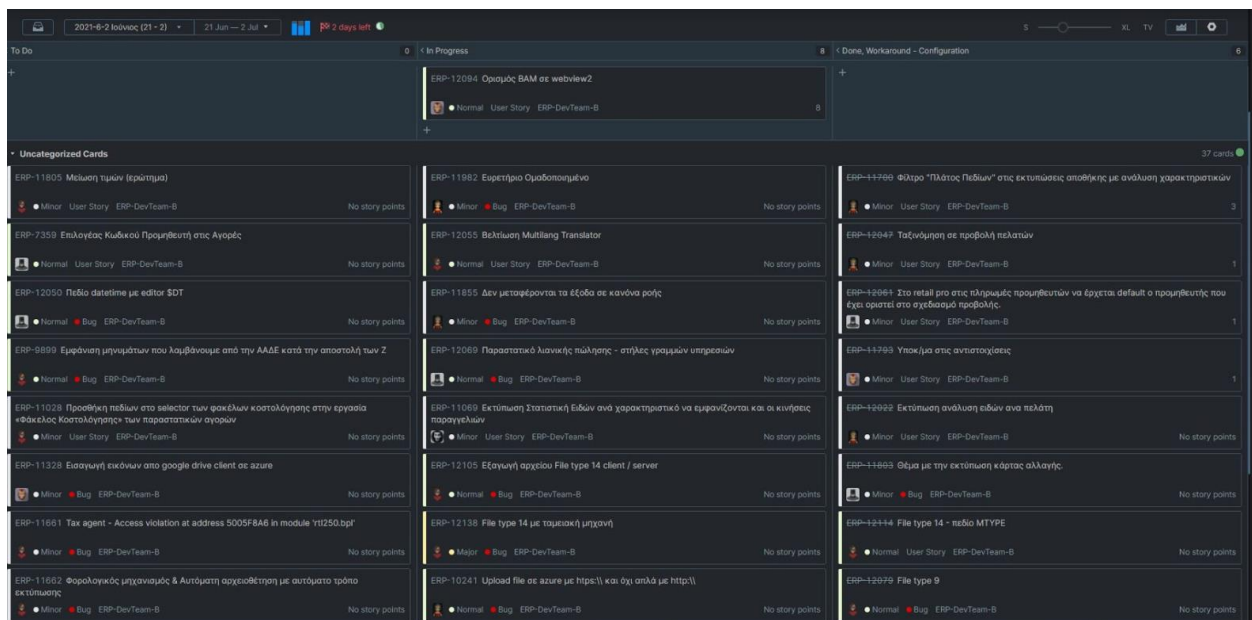
Αρχικά, απαιτείται χρόνος και από τον νέο εργαζόμενο να εκπαιδευτεί στις διαδικασίες και στον τρόπο λειτουργίας του τμήματος. Επίσης και κάποιο άλλο παραγωγικό μέλος της ομάδας αφιερώνει χρόνο προκειμένου να εκτελέσει χρέη εκπαιδευτή. Όπως γίνεται αντιληπτό η προσθήκη νέων μελών στο τμήμα απαιτεί αρκετό χρόνο προκειμένου αυτοί να συνεισφέρουν.

Αντίστοιχα όσον αφορά την επίλυση σφαλμάτων όσο ένας προγραμματιστής αποκτά εμπειρία στον τρόπο λειτουργίας τους τμήματος είναι ευκολότερο να επιλύσει κάποιο θέμα μόνος του χωρίς την συμμετοχή άλλου συναδέλφου. Επίσης όσο αναλαμβάνει την ανάπτυξη νέων προδιαγραφών έχει την επίβλεψη της συνολικής συντήρησης του συγκεκριμένου θέματος.

## Κεφάλαιο 5°

Στο κεφάλαιο αυτό γίνεται μια παρουσίαση των δεικτών απόδοσης (*Key Performance Indicators*) που αφορούν το τμήμα έρευνας και ανάπτυξης από τη σκοπιά της scrum μεθοδολογίας. Για την δημιουργία των συγκεκριμένων δεικτών KPIs έγινε συλλογή και ανάλυση δεδομένων από την πλατφόρμα καταγραφής και παρακολούθησης των user stories και των bugs. Ουσιαστικά πρόκειται για μια πλατφόρμα μέσω της οποίας γίνεται η καταγραφή των users stories και των σφαλμάτων που έχουν εντοπιστεί είτε από το εξωτερικό περιβάλλον είτε από το εσωτερικό. Ως εσωτερικό περιβάλλον ορίζονται τα δύο τμήματα που έχουν πρόσβαση στην πλατφόρμα δηλαδή αυτά της διαχείρισης προϊόντος και έρευνας & ανάπτυξης.

### 5.1 Συλλογή Δεδομένων Εσωτερικού Περιβάλλοντος



Εικόνα 11: Kanban Board, Youtrack

Τα δεδομένα που συλλέχθηκαν για τις ανάγκες δημιουργίας των δεικτών αφορούν τα τελευταία τρία χρόνια λειτουργίας της εταιρείας, κατά τα οποία εφαρμόζεται η scrum μέθοδος (2018-2020), και είναι τα εξής:

- Ο συνολικός αριθμός απαιτήσεων νέων προδιαγραφών (*user stories*) καθώς και μεγάλων απαιτήσεων (*epic*) που καταγράφηκε στην πλατφόρμα της εταιρείας από το εσωτερικό περιβάλλον και στα πλαίσια γενικής αναβάθμισης του προϊόντος ανά έτος.



- Ο αριθμός των *user stories* που υλοποιήθηκαν ανά έτος.
- Ο συνολικός αριθμός δυσλειτουργιών που καταγράφηκαν κατά τη διάρκεια της διαδικασίας ανάπτυξης ανά έτος.
- Ο συνολικός αριθμός σφαλμάτων - δυσλειτουργιών που επιλύθηκαν ανά έτος.
- Ο αριθμός ομάδων (*scrum teams*).
- Το συνολικό velocity των ολοκληρωμένων *user stories* στο τέλος κάθε έτους για το σύνολο των *scrum teams*.

Πρόκειται ουσιαστικά για αιτήματα που καταγράφονται από το τμήμα διαχείρισης προϊόντος. Αυτά σχετίζονται είτε με αιτήματα του εξωτερικού περιβάλλοντος (τόσο για προθήκη νέων δυνατοτήτων όσο και για επίλυση δυσλειτουργιών) είτε με απαιτήσεις που έχει θέσει η διοίκηση με σκοπό την εξέλιξη του προϊόντος.

Τα *user stories* και τα σφάλματα που είναι καταγεγραμμένα στη πλατφόρμα του εσωτερικού περιβάλλοντος δεν είναι ισάριθμα ένα προς ένα με τις καταγεγραμμένες απαιτήσεις για νέες προδιαγραφές και τα σφάλματα του εξωτερικού περιβάλλοντος. Όσον αφορά τις νέες προδιαγραφές δεν υπάρχει σαφής σχέση μεταξύ τους. Μια νέα απαίτηση από την πλευρά του πελάτη θα μπορούσε να είναι αρκετά απλή όσο ένα *user story* το οποίο θα υλοποιηθεί κατά τη διάρκεια ενός *sprint*, ωστόσο θα μπορούσε να είναι επιπέδου *epic* που να απαιτεί πολλά *user stories* σε διάστημα πολλών *sprints* μέχρι να υλοποιηθεί.

	2018	2019	2020
Ομάδες Scrum	2	2	3
Απαιτήσεις επιπέδου Epic	20	22	35
Απαιτήσεις επιπέδου User Story	1028	1179	1328
Υλοποιημένα User Stories	715	895	1042
Καταγεγραμμένες δυσλειτουργίες	886	790	720
Δυσλειτουργίες που επιλύθηκαν	749	683	655
Ικανότητα (Velocity)	3750	4150	6375

Πίνακας 2: Δεδομένα εσωτερικού περιβάλλοντος

## 5.2 Ποσοτικοί δείκτες απόδοσης(KPIs)

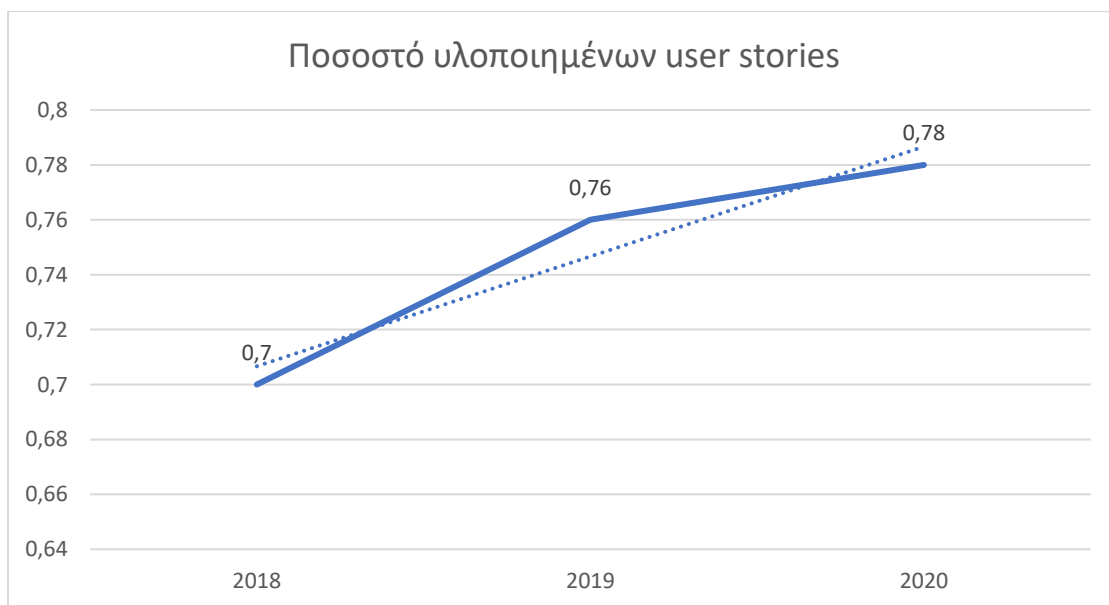
### 5.2.1 Ποσοστό user stories που υλοποιήθηκαν

#### 5.2.1.1 Ανάγκη

Όπως ήδη έχει αναφερθεί στόχος της εφαρμογής της μεθόδου agile είναι η αύξηση της αποδοτικότητας και του καλύτερου ελέγχου σχετικά με την ανάπτυξη και την συντήρηση του προϊόντος. Μέσω της ανάλυσης του ποσοστού των user stories που ολοκληρώνονται μπορούν να βγουν συμπεράσματα που αφορούν την εξέλιξη και βελτίωση του προσωπικού του τμήματος έρευνας και ανάπτυξης. Μπορεί να γίνει καλύτερη εκτίμηση του φόρτου εργασίας που θα απαιτηθεί και αντίστοιχα να φτάσει σε στάδιο ωριμότητας κάθε ομάδα ώστε να μπορεί να εκτιμήσει με ακρίβεια τα user stories που θα ολοκληρώνει στη διάρκεια κάθε sprint.

#### 5.2.1.2 Ανάλυση

Ο συγκεκριμένος δείκτης υπολογίζεται από το κλάσμα των υλοποιημένων users stories στο σύνολο των scrum teams ως προς τον αριθμό των εισαγόμενων user stories από το τμήμα διαχείρισης προϊόντος στο τέλος του κάθε έτους.



Διάγραμμα 8: Ποσοστό υλοποιημένων user stories

### 5.2.1.3 Συμπεράσματα

Όπως φαίνεται από το διάγραμμα 8 το ποσοστό των user stories που υλοποιούνται επιτυχώς και παραδίδονται στους product owners αυξάνεται με σχετικά σταθερή τάση. Αυτό θα επιτρέψει μελλοντικά καλύτερη εκτίμηση την απαιτούμενης προσπάθειας και κατ' επέκταση καλύτερο προγραμματισμό των εργασιών.

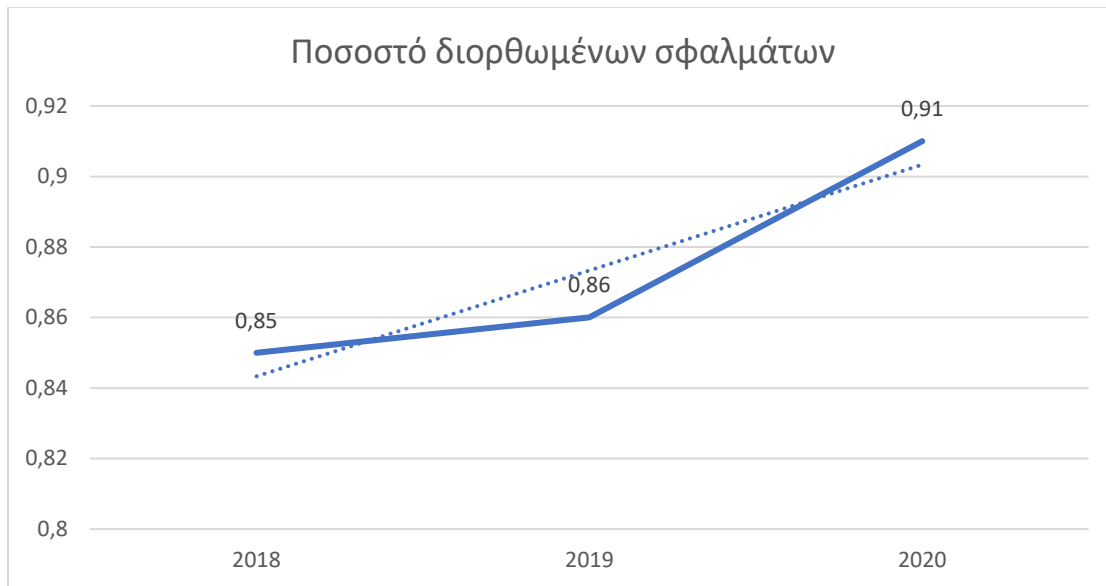
## 5.2.2 Ποσοστό δυσλειτουργιών που διορθώθηκαν

### 5.2.2.1 Ανάγκη

Τα σφάλματα που διορθώνονται εντός της ομάδας ανάπτυξης δεν είναι απαραίτητα σφάλματα που έχουν καταλήξει στον πελάτη. Όσα περισσότερα σφάλματα διορθώνονται στο στάδιο του ποιοτικού ελέγχου του προϊόντος τόσο περισσότερο διασφαλίζεται ότι οι πελάτες της εταιρείας λαμβάνουν ένα προϊόν αξιόπιστο που μπορούν να χρησιμοποιούν απροβλημάτιστα.

### 5.2.2.2 Ανάλυση

Ο συγκεκριμένος δείκτης υπολογίζεται από το κλάσμα των διορθωμένων σφαλμάτων στο σύνολο των scrum teams ως προς το σύνολο των σφαλμάτων που καταγράφεται από το τμήμα διαχείρισης προϊόντος ή και από σφάλματα που εντοπίζονται κατά τη διάρκεια ελέγχου πριν νέες προδιαγραφές φτάσουν στον πελάτη. Τα σύνολα αφορούν το τέλος του κάθε έτους.



*Διάγραμμα 9: Ποσοστό διορθωμένων σφαλμάτων*

### 5.2.2.3 Συμπεράσματα

Όπως παρατηρείται η πλειοψηφία των αναφερόμενων σφαλμάτων επιλύεται από την ομάδα προγραμματιστών. Με το πέρασμα των ετών βελτιώνεται σταθερά η απόδοση όσον αφορά την επίλυση σφαλμάτων. Αισιόδοξα θα μπορούσε μελλοντικά να φτάσει και το 100% σύμφωνα με την τάση του διαγράμματος 9.

## 5.2.3 Users Stories – Σφάλματα ανά Scrum Team

### 5.2.3.1 Ανάγκη

Βασική οντότητα της όλης μεθοδολογίας είναι η ομάδα. Λειτουργεί σχετικά αυτόνομα και καθορίζει μόνη της τι θα υλοποιήσει βάσει των δυνατοτήτων των μελών της.

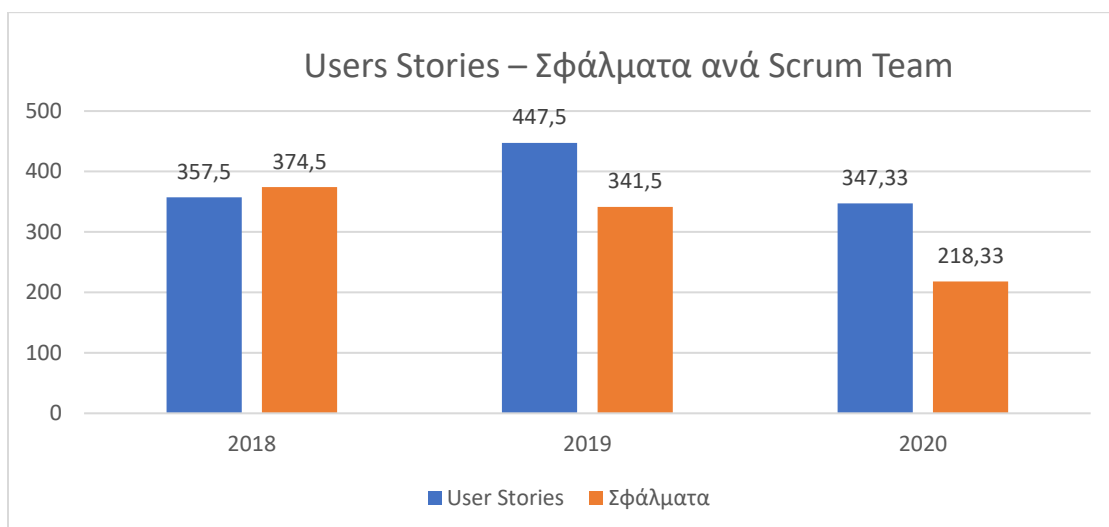
Μέσα από τη μέθοδο agile κάθε ομάδα ωριμάζει. Τα μέλη της ομάδας αλληλοβοηθούνται και αυτό έχει σαν αποτέλεσμα η κάθε ομάδα σαν σύνολο να αυξάνει τις ικανότητές της τόσο σε τεχνικό επίπεδο αποκτώντας τεχνογνωσία και μεταφέροντας τη γνώση στο σύνολο της ομάδας όσο και σε επίπεδο οργάνωσης. Αυτό έχει σαν αποτέλεσμα η ομάδα να αυξάνει την

αποτελεσματικότητά της και ταυτόχρονα να αποκτά κριτική άποψη σχετικά με τα user stories που θα δεσμευτεί να υλοποιήσει κατά τη διάρκεια του sprint.

Θα ήταν ωφέλιμο από τη πλευρά της διοίκησης κάθε εταιρείας που εφαρμόζει το agile μοντέλο να μπορεί να εκτιμήσει τη δυναμική κάθε ομάδας και βάσει των προδιαγραφών που έχει θέσει για μια επιτυχημένη παραγωγική διαδικασία να κάνει της ανάλογες παρεμβάσεις.

### 5.2.3.2 Ανάλυση

Ο συγκεκριμένος δείκτης υπολογίζεται από το σύνολο των υλοποιημένων user stories και των επιδιορθωμένων σφαλμάτων σε σχέση με τις αντίστοιχες εισαγωγές στη πλατφόρμα παρακολούθησης στο τέλος κάθε έτους και σε σχέση με το πλήθος των scrum ομάδων.



Διάγραμμα 10: Μέσος όρος Ολοκληρωμένων users stories-σφαλμάτων ανά ομάδα

### 5.2.3.3 Συμπεράσματα

Στα πλαίσια της εργασίας δεν αναλύθηκε περαιτέρω η απόδοση κάθε ομάδας ξεχωριστά για λόγους προστασίας δεδομένων και αποφυγής σύγκρισης των ομάδων μεταξύ τους. Το συμπέρασμα που προκύπτει ωστόσο από το διάγραμμα 10 είναι πως τη πρώτη χρονιά εφαρμογής της μεθόδου κάθε ομάδα αφιέρωσε περισσότερο χρόνο για την επίλυση σφαλμάτων. Τα σφάλματα

αυτά είναι πιθανό να προέκυψαν κυρίως κατά τη διαδικασία ανάπτυξης νέων δυνατοτήτων όσο ακόμα οι ομάδες προσπαθούσαν να οργανωθούν στο νέο τρόπο λειτουργίας.

## 5.2.4 Απόδοση των Scrum Ομάδων ανά sprint

### 5.2.4.1 Ανάγκη

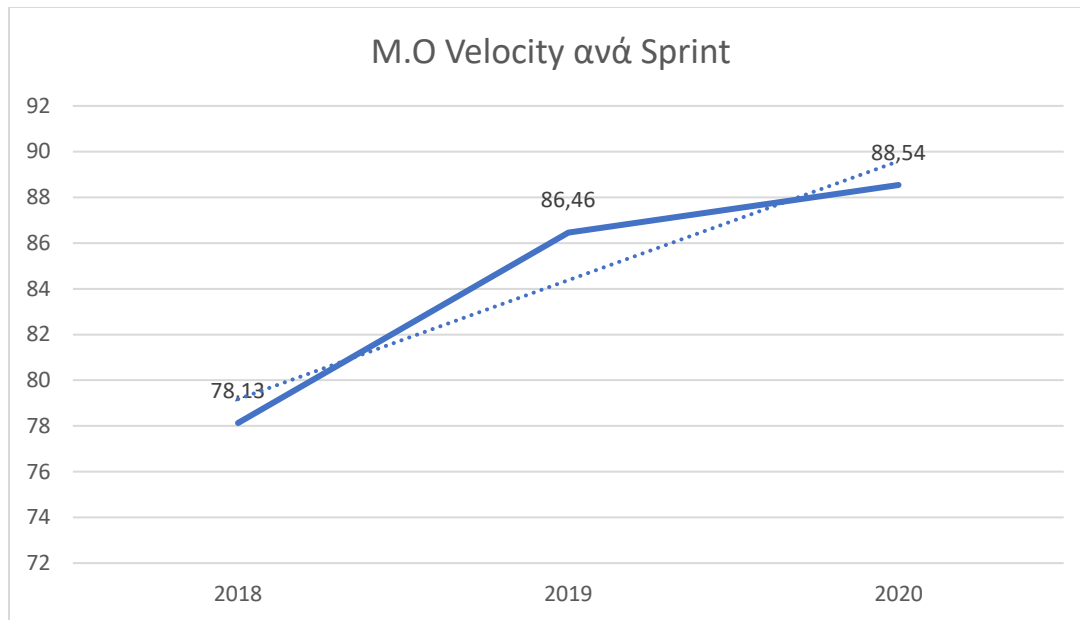
Ένας άλλος τρόπος ελέγχου της απόδοσης κάθε ομάδας αφορά το velocity της. Στην ουσία πρόκειται για τον συνολικό βαθμό δυσκολίας των user stories που καλείται να φέρει εις πέρας σε κάθε sprint.

Σε αντίθεση με το πλήθος των user stories που υλοποιούνται σε κάθε ομάδα άξιος αναφοράς είναι και ο βαθμός δυσκολίας των user stories αυτών. Μπορεί μια ομάδα να αφιερώσει ένα ολόκληρο sprint για να φέρει εις πέρας δύο user stories την ώρα που μια άλλη ομάδα μπορεί να ολοκληρώσει δέκα. Ωστόσο αν ο βαθμός δυσκολίας των 2 user stories της πρώτης ομάδας είναι αθροιστικά μεγαλύτερος από το άθροισμα των δέκα users stories της δεύτερης ομάδας τότε θεωρητικά η πρώτη ομάδα είναι πιο αποτελεσματική.

Όσο πιο έμπειρα στελέχη έχει μια ομάδα τόσο μεγαλύτερο velocity έχει. Επιπλέον όσο καλύτερη διαχείριση της γνώσης μπορεί να πραγματοποιήσει βοηθώντας και πιο άπειρα μέλη μπορεί να επιτύχει ακόμα μεγαλύτερη αύξηση του velocity της.

### 5.2.4.2 Ανάλυση

Ο συγκεκριμένος δείκτης υπολογίζεται από το σύνολο του velocity που έχει καταγραφεί στην πλατφόρμα διαχείρισης των user stories σε σχέση με το πλήθος ομάδων και το πλήθος των sprints ανά έτος. Στα πλαίσια των μετρήσεων θεωρούμε ότι ο αριθμός των sprints ανά έτος είναι 24. Αυτό προκύπτει από τη διάρκεια του sprint που είναι δύο εβδομάδες καθώς και του ότι μέσα στη καλοκαιρινή περίοδο δύο sprints δεν πραγματοποιούνται.



Διάγραμμα 11: Μέσος όρος velocity ανά sprint

#### 5.2.4.3 Συμπεράσματα

Όπως και στον προηγούμενο δείκτη δεν αναλύθηκε περαιτέρω η απόδοση κάθε ομάδας ξεχωριστά για λόγους προστασίας δεδομένων και αποφυγής σύγκρισης των ομάδων μεταξύ τους. Το συμπέρασμα που προκύπτει είναι ότι όσο περισσότερο διάστημα εφαρμόζεται η μεθοδολογία agile τόσο αυξάνεται η ικανότητα των μελών των ομάδων. Ωστόσο, σύμφωνα και με την τάση του γραφήματος 11 μπορούμε να υποθέσουμε ότι αν και εφόσον δεν γίνουν αλλαγές στις ομάδες με προσθήκες νέων μελών και αποχωρήσεις παλαιότερων και πιο έμπειρων στελεχών το velocity μετά από ένα διάστημα θα σταθεροποιηθεί σε κάποιο σημείο. Σύμφωνα με το επίπεδο ικανότητας αυτό θα μπορεί η εταιρεία να υπολογίζει το κόστος ανάπτυξης των νέων προδιαγραφών.

## Κεφάλαιο 6°

### 6.1 Συμπεράσματα

Η agile μέθοδος θεωρείται ιδανική για έργα πληροφορικής, μέσα από τη μελέτη περίπτωσης της Softone προκύπτει ότι μπορεί να βρει εφαρμογή και σε τμήματα ανάπτυξης και συντήρησης ενός προϊόντος. Όσο πιο σύνθετη είναι η διαδικασία ανάπτυξης και συντήρησης ενός προϊόντος όπως είναι ένα ERP, τόσο πιο αποδοτική θα είναι η εφαρμογή της μεθόδου agile.

Στα πλαίσια της παρούσας μελέτης παρουσιάστηκαν οι ρόλοι καθώς και οι διαδικασίες που θέσπισε η εταιρεία στα πλαίσια της υιοθέτησης της agile μεθοδολογίας. Για την εφαρμογή της μεθόδου χρειάστηκε να γίνει σχετική εκπαίδευση όλων των εμπλεκόμενων στελεχών, να γίνει αναδιοργάνωση του τμήματος έρευνας και ανάπτυξης και διαχείρισης προϊόντος προκειμένου να οριστούν άτομα με συγκεκριμένους ρόλους και τέλος πραγματοποιήθηκαν απαραίτητες αλλαγές στη διαδικασία ανάθεσης εργασιών στα μέλη τους. Στα πλαίσια των αλλαγών δημιουργήθηκαν ομάδες, αποτελούμενες από έμπειρα και νεότερα στελέχη. Αυτό είχε σαν αποτέλεσμα τον καλύτερο διαμερισμό και την ευκολότερη και πιο οργανωμένη μετάδοση γνώσης και τεχνογνωσίας εντός των ομάδων. Ταυτόχρονα όλα τα μέλη κλήθηκαν να πάρουν πρωτοβουλίες και να κάνουν προτάσεις, γεγονός που οδήγησε στο να ωριμάσουν ταχύτερα αυξάνοντας την αποδοτικότητα των ομάδων τους και συνολικά όλου του τμήματος.

Με την εφαρμογή της agile μεθοδολογίας έγινε πολύ πιο εύκολη η καθημερινή επικοινωνία, η επίλυση προβλημάτων και η παρακολούθηση των εκκρεμοτήτων και συνολικά αποτελεσματικότερος ο έλεγχος εντός του τμήματος με την πραγματοποίηση των καθημερινών συναντήσεων (daily stand up meetings). Παράλληλα, η πραγματοποίηση των sprint review συνέβαλε στην καλύτερη επικοινωνία του τμήματος έρευνας και ανάπτυξης με τα υπόλοιπα τμήματα και κατ' επέκταση αυτό είχε θετικό αντίκτυπο στην ικανοποίηση του τελικού πελάτη. Μέσα από τα sprint reviews επίσης ήταν άμεση η ανατροφοδότηση για νέα projects. Δόθηκε η ευκαιρία στους άμεσα ενδιαφερόμενους να κάνουν προτάσεις για αλλαγές και βελτιώσεις ούτως ώστε οι εργατοώρες που θα αφιέρωναν οι προγραμματιστές να είναι αποδοτικές και να περιοριστεί η άσκοπη ανάπτυξη.



Όπως προέκυψε από την ανάλυση των δεδομένων της πλατφόρμας καταγραφής αιτημάτων του εξωτερικού περιβάλλοντος, το επίπεδο παραγωγής παρουσίασε σημαντική βελτίωση. Αυτό γίνεται ξεκάθαρο βλέποντας τους σχετικούς δείκτες για το διάστημα των τελευταίων έξι ετών λειτουργίας της εταιρείας. Το ποσοστό των υλοποιημένων προδιαγραφών ανά εγκατάσταση αυξήθηκε και λαμβάνοντας υπόψη ότι μέσα σε μία εξαετία υπήρξε τεράστια αύξηση των νέων εγκαταστάσεων, είναι φανερό το πόσο πιο αποδοτικά λειτούργησε το τμήμα. Επιπλέον μειώθηκαν σημαντικά τα καταγεγραμμένα σφάλματα που έφτασαν στον τελικό πελάτη, γεγονός που υποδηλώνει ότι το πρόγραμμα ωριμάζει και γίνεται όλο και πιο αξιόπιστο. Τέλος αυξήθηκε η αποτελεσματικότητα κάθε προγραμματιστή ανεξαρτήτως από το μέγεθος του τμήματος σε κάθε χρονιά.

Στα πλαίσια της μελέτης διερευνήθηκαν και τα δεδομένα που αφορούν στην διαχείριση των αιτημάτων από τη σκοπιά της agile μεθοδολογίας. Μέσα στα τρία χρόνια που εφαρμόζεται η μέθοδος παρουσιάστηκε αύξηση των υλοποιημένων user stories για κάθε scrum ομάδα. Αυτό από μόνο του αποτελεί απόδειξη ότι όσο ωριμάζει η εταιρεία στη εφαρμογή της μεθοδολογίας, οι ομάδες αποκτούν ικανότητα αξιολόγησης των uses stories και οι δεσμεύσεις που δίνουν για τα παραδοτέα πραγματοποιούνται. Επομένως μπορεί να γίνει καλύτερη εκτίμηση σχετικά με την απαιτούμενη προσπάθεια που θα καταβάλει το τμήμα και αντίστοιχα καλύτερος χρονοπρογραμματισμός για την υλοποίηση νέων προσθηκών - δυνατοτήτων. Παράλληλα όπως και με αντίστοιχο δείκτη από το εξωτερικό περιβάλλον βλέπουμε σημαντική αύξηση της ικανότητας των στελεχών του τμήματος ώστε να μπορούν να υλοποιούν περισσότερες απαιτήσεις αλλά να πραγματοποιούν και επιδιορθώσεις αυξάνοντας την αξιοπιστία του προγράμματος.

Συνεπώς αξιολογώντας τα αποτελέσματα των δεικτών που μελετήθηκαν θα μπορούσαμε με ασφάλεια να πούμε ότι η εφαρμογή της μεθοδολογίας στο τμήμα έρευνας και ανάπτυξης έπαιξε καθοριστικό ρόλο στην αύξηση της αποδοτικότητάς του. Όσο η εταιρεία αναπτύσσεται και μεγαλώνει θα αυξάνονται και οι απαιτήσεις του πελατολογίου της, ωστόσο με μια ευέλικτη διαχείριση των θεμάτων θα συνεχίσει να ανταποκρίνεται με συνέπεια προσφέροντας ένα πλήρες και αξιόπιστο προϊόν με πολλές δυνατότητες ικανό να ανταπεξέλθει στον ανταγωνισμό.

Τέλος, το βασικότερο πλεονέκτημα που προσέθεσε η συνολική εφαρμογή της μεθόδου στην εταιρεία και που δεν είναι εύκολο να παρουσιαστεί μέσα από την ανάλυση των συγκεκριμένων δεδομένων είναι η διαφάνεια στις διαδικασίες ανάπτυξης.

## 6.2 Μελλοντική μελέτη - προτάσεις βελτίωσης

Όσο καλύτερα εφαρμόζεται η agile μεθοδολογία στην εταιρεία τόσο πιο ρεαλιστικά αποτελέσματα θα εμφανίζονται στους δείκτες απόδοσης. Επίσης θα είναι δυνατή η δημιουργία περισσότερων δεικτών οι οποίοι θα βοηθήσουν σε ακόμα καλύτερο έλεγχο.

Ενδιαφέρον θα είχε μελλοντικά να μελετηθεί η σχέση μεταξύ των απαιτήσεων των πελατών (new specs) με τα epics και τα user stories. Επίσης θα μπορούσε να γίνει υπολογισμός των δεικτών για κάθε scrum ομάδα προκειμένου να ελεγχθεί αν υπάρχει ισορροπία στο τμήμα ή αν πρέπει να γίνουν διορθωτικές κινήσεις. Επιπλέον, σημαντική θα ήταν η παρακολούθηση του χρόνου που χρειάζεται προκειμένου να ολοκληρωθεί ένα user story ή epic αντίστοιχα με απώτερο στόχο την παρακολούθηση της εξυπηρέτησης των πελατών και αν τηρείται το συμφωνηθέν επίπεδο υπηρεσιών (service level agreement).



## Βιβλιογραφία

Rasmusson, J. (2010). *The agile samurai: How agile masters deliver great software*. Pragmatic Bookshelf.

Fowler, M., & Highsmith, J. (2001). *The agile manifesto*. *Software development*, 9(8), 28-35.

Cockburn, A. (2004). *Crystal clear: A human-powered methodology for small teams: A human-powered methodology for small teams*. Pearson Education.

Fowler, M., & Lewis, J. (2015). *Microservices: Nur ein weiteres Konzept in der Softwarearchitektur oder mehr*. *Objektspektrum*, 1(2015), 14-20.

Jeffries, R. (2001). *What is extreme programming*. *XP magazine*, 11.

Sverrisdottir, H. S., Ingason, H. T., & Jonasson, H. I. (2014). *The role of the product owner in scrum-comparison between theory and practices*. *Procedia-Social and Behavioral Sciences*, 119, 257-267.

Judy, K. H., & Krumins-Beens, I. (2008, January). *Great scrums need great product owners: Unbounded collaboration and collective product ownership*. In *Proceedings of the 41st Annual Hawaii International Conference on System Sciences (HICSS 2008)* (pp. 462-462). IEEE.

Stare, A. (2014). *Agile project management in product development projects*. *Procedia-Social and Behavioral Sciences*, 119, 295-304.

Dingsøyr, T., Nerur, S., Balijepally, V., & Moe, N. B. (2012). *A decade of agile methodologies: Towards explaining agile software development*.

Γεωργιάδου, Α. (2014). *Διερεύνηση της χρήσης ευέλικτων μεθόδων διοίκησης έργου*.

Kumar, G., & Bhatia, P. K. (2012). *Impact of agile methodology on software development process*. *International Journal of Computer Technology and Electronics Engineering (IJCTEE)*, 2(4), 46-50.

Schwaber, K., & Beedle, M. (2002). *Agile software development with Scrum* (Vol. 1). Upper Saddle River: Prentice Hall.

The agile manifesto explained [Online]

*Available at:* <https://laptrinhx.com/the-agile-manifesto-explained-1018380669/>

ADS Practices [Online]

*Available at*

[https://www.tutorialspoint.com/adaptive\\_software\\_development/adaptive\\_software\\_development\\_practices.htm](https://www.tutorialspoint.com/adaptive_software_development/adaptive_software_development_practices.htm)

Schwaber M, & Sutherland J, 2013. The scrum guide.

*Available at:* <https://www.scrum.org/scrum-guide>