

Risk-assessment-v0.1

Κωδικός: Risk-assessment

Έκδοση: v0.1

ΣΥΝΕΡΓΑΖΟΜΕΝΑ ΜΕΛΗ

- | | | | |
|----------------------|-------------|---------------------|---------------|
| ➤ Παπαδοπούλου Μαρία | AM: 1072494 | 4 ^ο έτος | Peer reviewer |
| ➤ Σφήκας Θοδωρής | AM: 1072550 | 4 ^ο έτος | Author |

Risk Management

Η αξιολόγηση, εκτίμηση και ομαδοποίηση του risk assessment, πραγματοποιείται με βάση ενός συνδυασμού των προτροπών των Sommerville και Boehm.

Risk Identification :

Risk	Affects	Description
1. Αλλαγή στα μέλη συνεργασίας	Project	Αλλαγή της σύστασης της ομάδας διεκπεραίωσης του project.
2. Προβληματικοί προσωπικοί υπολογιστές (hardware malfunctions)	Project	Η αδυναμία κατοχής ενός λειτουργικού υπολογιστή καθιστά την συνεισφορά ενός μέλους της ομάδας σχεδόν αδύναμη.
3. Μικρή εμπειρία μελών σε software development	Project And Product	Κάθε ένα μέλος της ομάδας θα μπορούσε να θεωρηθεί ένας junior developer χωρίς εργασιακή εμπειρία.
4. Καθυστερήσεις υλοποίησης	Project and product	Καθυστέρηση υλοποίησης αναγκαίων υλοποιήσεων και deadline misses. Λάθος χρονοπρογραμματισμοί.
5. Υπερίσχυση του Ανταγωνισμού	Business	Ο κύριος ανταγωνισμός του project ξεπερνά σε ταχύτητα και απόδοση την υλοποίηση της ομάδας. Lose of market leading position.
6. Αλλαγή των κύριων τεχνολογιών	Business	Η υποτυπώδεις τεχνολογίες του χρησιμοποιούνται ως κορμός της υλοποίησης καθίστανται απαρχαιωμένες
7. Αδυναμία χρήσης software λόγω licensing	Business and Product	Open Source and ιδιαίτερα licenses μπορεί να εμποδίζουν τον χρηματισμό του προϊόντος με την χρήση τους.
8. Έλλειψη χρηματοδότησης	Business	Αδυναμία εύρεσης απαραίτητου επενδυτικού κεφαλαίου και χρηματοδότησης για την πληρωμή της ομάδας σε περίπτωση αδυναμίας dilution του μετοχικού κεφαλαίου της υπό σύστασης start-up.
9. Υποεκτίμηση χρόνου των bug fixes	Project	Καθυστέρηση της υλοποίησης λόγω αδιαφανών παραγόντων και αδυναμία σύστασης λειτουργικού λογισμικού.
10. Αδυναμία εργασίας των developers	Project	Αδειοδότηση λόγω ισχυρών ασθενειών, σημαντικών προσωπικών παραγόντων και κυοφορήσης.
11. Υποεκτίμηση του μεγέθους δουλειάς	Project	Το μέγεθος του συστήματος υλοποίησης είναι πολύ μεγαλύτερο από ότι αναμένουμε.
12. Αλλαγές στις προϋποθέσεις	Project and Product	Μεγάλος αριθμός αναγκαίων αλλαγών των βασικών υποσυστημάτων, των απαιτήσεων και των ιδεών προς υλοποίηση.
13. Υπερεκτίμηση των δυνατοτήτων της ομάδας	Project and Product	Αδυναμία με βάση το παρόν επίπεδο δεξιοτήτων των μελών της ομάδας να καλύψουν ικανοποιητικά τις ανάγκες του λογισμικού.
14. Αδυναμία beta testing	Project and Product	Το λογισμικό δεν μπορεί να ελεγχθεί για την λειτουργικότητα του και δεν υπάρχει πρόταση συστάσεων από κλειστό beta testing cycle.

15, Ανικανότητα εργασίας με Scrumban	Project	Αδυναμία ορθής συνεργασίας μέσω συνδυασμού μοντέλων Scrum και Kanban λόγω απειρίας.
16. Unscalable βάση δεδομένων	Product	Η βάση δεδομένων που σχεδιάσαμε και υλοποιήσαμε δεν επαρκεί για τον όγκο των δεδομένων της deployed εφαρμογής.
17. Προβλήματα ασφάλειας	Product	Αναφερόμενοι σε web application υπάρχουν κίνδυνοι ασφάλειας και προστασίας προσωπικών δεδομένων.
18. Αδυναμία των servers να ανταποκριθούν στο workload	Business and Product	Σε περίπτωση μεγάλης επεκτασιμότητας οι hostings servers που χρησιμοποιούμε μπορεί να υπερφορτωθούν.

Πίνακας 1. Αναγνώριση ρίσκων, παρουσίασή τους και κατηγοριοποίηση βάση του project aspect που αφορούν.

Ομαδοποίηση των παραπάνω παραγόντων πιθανών κινδύνων βάση του τύπου ρίσκου :

Risk type	Possible Risks
Estimation	4. Καθυστερήσεις υλοποίησης 9. Υποεκτίμηση χρόνου των bug fixes
Organizational	8. Έλλειψη χρηματοδότησης 14. Αδυναμία beta testing 15. Ανικανότητα εργασίας με Scrumban
People	1. Αλλαγή στα μέλη συνεργασίας 3. Μικρή εμπειρία μελών σε software development 10. Αδυναμία εργασίας των developers 13. Υπερεκτίμηση των δυνατοτήτων της ομάδας
Requirements	12. Αλλαγές στις προϋποθέσεις
Technology	5. Υπερίσχυση του Ανταγωνισμού 6. Αλλαγή των κύριων τεχνολογιών 16. Unscalable βάση δεδομένων 17. Προβλήματα ασφάλειας
Tools	2. Προβληματικοί προσωπικοί υπολογιστές 7. Αδυναμία χρήσης software λόγω licensing 18. Αδυναμία των servers να ανταποκριθούν στο workload

Πίνακας 2. Κατηγοριοποίηση ρίσκων βάση του είδους τους.

Risk Analysis :

Κατηγοριοποίηση των ρίσκων βάση της πιθανότητας εμφάνισης τους και της κρισιμότητας τους :

Risk	Probability	Effect Consequence
1. Αλλαγή στα μέλη συνεργασίας	Low	High
2. hardware malfunction	Low	Moderate
3. Μικρή εμπειρία μελών σε software development	High	High
4. Καθυστερήσεις υλοποίησης	High	Moderate
5. Βελτιστοποίηση του Ανταγωνισμού	Moderate	High
6. Αλλαγή των κύριων τεχνολογιών	Low	High
7. Αδυναμία χρήσης software λόγω licensing	Moderate	Moderate
8. Έλλειψη χρηματοδότησης	Low	Low
9. Υποεκτίμηση χρόνου των bug fixes	High	High
10. Αδυναμία εργασίας των developers	Low	High
11. Υποεκτίμηση του μεγέθους δουλειάς	High	Moderate
12. Αλλαγές στις προϋποθέσεις	High	Moderate
13. Υπερεκτίμηση των δυνατοτήτων της ομάδας	High	High
14. Αδυναμία beta testing	Low	High
15. Ανεκτικότητα εργασίας με Scrumban	High	High
16. Unscalable βάση δεδομένων	Low	High
17. Προβλήματα ασφάλειας	Low	High
18. Αδυναμία των servers να ανταποκριθούν στο workload	Low	Moderate

Πίνακας 3. Καθορισμός πιθανότητας εμφάνισης και σημαντικότητας συνεπειών κάθε αναγνωρισμένου ρίσκου.

Risk Planning :

Βάση της παραπάνω ανάλυσης ρίσκου επιλέγουμε τις περιπτώσεις με την μεγαλύτερη σοβαρότητα για την υλοποίηση του έργου μας με στόχο την διαμόρφωση στρατηγικών και μεθόδων αντιμετώπισης τους.

Risk	Strategy
3. Μικρή εμπειρία μελών σε software development	Avoidance Strategy: Καθώς κατανοούμε την θέση νέων προγραμματιστών, δεν κατέχουμε ούτε την απαραίτητη γνώση ούτε την πρακτική εμπειρία για να εκτελέσουμε έργα μεγάλων διαστάσεων και ομάδων. Για την αποφυγή λαθών αποφασίζουμε να εξετάζουμε ιδιαιτέρως κάθε προς χρήση λογισμικό και τους στόχους μας. Η ταυτόχρονη εργασία μας με βάση ένα Scrum μοντέλο και τα daily meetings διευκολύνουν την επικοινωνία και την έκφραση ενδοιασμών.
4. Καθυστερήσεις υλοποίησης	Minimization Strategy: Γνωρίζουμε πως όπως και στα περισσότερα projects θα υπάρξουν πολλαπλές καθυστερήσεις. Συνυπολογίζοντας την εμπειρία μας είναι σχεδόν αδύνατο να κρίνουμε ορθά τις χρονικές απαιτήσεις των υλοποιήσεων. Αποφασίσαμε πως καθώς λειτουργούμε παράλληλα σε πολλαπλά κομμάτια κώδικα μέσα στα πλαίσια του agile programming σε περίπτωση αργοπορίας ενός branch, ο scrum master οφείλει να βοηθάει και να συνεισφέρει στην καθυστερημένη διαδικασία προτού αυτή ξεπεράσει τα deadlines παράδοσης.
9. Υποεκτίμηση χρόνου των bug fixes	Minimization Strategy: Η χρήση του Scrum ως development model απαιτεί την δημιουργία ημι-αυτόνομων προϊόντων και υποσυστημάτων στο τέλος του κάθε sprint cycle. Λόγω των πολλαπλών cycles ο κώδικας σε κάθε subsystem καθώς και η δυνατότητα συγχώνευσης των subsystems εξετάζονται συνεχώς.
11. Υποεκτίμηση του μεγέθους δουλειάς	Contingency Plan: Λειτουργώντας με βάση την τμηματοποίηση του project backlog σε μικρότερα υποσυστήματα σε κάθε Sprint cycle εκμεταλλευόμενοι τον συνδυασμό των μοντέλων Scrum και Kanban, όπως αναφέρουμε στο project plan, Εξετάζουμε μετρήσιμα τον χρόνο που απαιτεί κάθε use-case. Σε περίπτωση που παρατηρείται αδυναμία κάλυψης των στόχων του έργου σε κάθε Sprint Cycle, μπορούμε να οδηγηθούμε σε μείωση των project requirements με στόχο την παράδοση ενός ολοκληρωμένου προϊόντος με κάποια ελλειπή δυνατότητα. Ύστερα από την παράδοση του λογισμικού μπορούμε να εισάγουμε εκ νέου την λειτουργικότητα κατά την διαδικασία του project maintenance.
12. Αλλαγές στις προϋποθέσεις	Minimization Strategies: Η πιθανότητα αλλαγών και επανασχεδιασμού των προϋποθέσεων του έργου είναι

	<p>τεράστια. Για την μείωση απρόβλεπτων αλλαγών σε κάθε κυκλική διαδικασία του Scrum επανεξετάζουμε την δυνατότητα υλοποίησης και τις περαιτέρω λειτουργικότητες που παθανών επαυμούμε. Μία start-up εταιρία με ισομερισμένο μετοχικό κεφάλαιο δεν επικοινωνεί με κάποιον εξωτερικό πελάτη ή διαχειριστή. Το έργο αποτελείται και σχηματίζεται με βάση τις αντιλήψεις της ομάδας και με την προϋπόθεση συνεχής επικοινωνίας και καθαρών στόχων μπορούμε να δεσμευτούμε για συγκεκριμένες λειτουργικότητες.</p>
<p>14. Ανικανότητα εργασίας με Scrumban</p>	<p>Minimization Strategies: Γνωρίζουμε πως λόγω της έλλειψης εμπειρίας μεταξύ των μελών της ομάδας είναι πολύ παθανό να μην κατανοείται επαρκώς το μοντέλο συνεργασίας. Συνυπολογίζοντας ακόμα πως η χρήση του μοντέλου της Kanban αφορά περισσότερο experienced engineers μπορεί να μην αποδόσει τα αποτελέσματα που περιμένουμε. Λόγω των παραπάνω σε κάθε Sprint Cycle αποφασίζουμε να κάνουμε review τον Scrum Master για την συνεχή βελτίωση της οργάνωσης. Καθώς χρησιμοποιούμε το Kanban όχι για ολόκληρη την υλοποίηση του project αλλά αποκλειστικά για κάθε Sprint Cycle τόσο το λιγότερο φόρτο εργασιών καθώς και η συνεχής επικοινωνία των μελών περιμένουμε να μειώσει παθανές δυσκολίες και αστοχίες μας.</p>

Πίνακας 4. Παρουσίαση τεχνικών αντιμετώπισης σημαντικότερων ρίσκων.

Εργαλεία που χρησιμοποιήθηκαν :

Για την συγγραφή της προσκείμενης αναφοράς χρησιμοποιήθηκαν τα Only Office. Ένα Open sourced alternative του Microsoft Word.