

v1.0

ΠΕΡΙΕΧΟΜΕΝΑ

1 Σύνθεση Ομάδας.....	3
2 Αλλαγές	5
3 Μοντέλο – Αρμοδιότητες	6
4 Χρονοπρογραμματισμός	8
5 Κατανομή Προσπάθειας και Συμπεράσματα	13
6 Εργαλεία	14

1 Σύνθεση Ομάδας

1.1 Μέλη

Meet The Team

Behind The Creation of **ReadEase**



Γεώργιος Καραγιάννης
1084586
REVIEWER



Ευαγγελία Κολάγκη
1084599
STYLISTICS EDITOR



Μαρία Κουρή
1084526
REVIEWER



Μηλιτιάδης Μαντές
1084661
AUTHOR



Χρυσαιγή Πατέλη
1084513
CO-AUTHOR

1.2 Λόγοι Σύνθεσης

Έχοντας ήδη συνεργαστεί ξανά στο παρελθόν με επιτυχία, γνωρίζαμε ήδη ως ομάδα τις δεξιότητες και δυνατότητες του κάθε μέλους. Επομένως, η συνεργασία μεταξύ μας θεωρούμε ότι θα είναι αποδοτική και θα φέρει σε ικανοποιητικό βαθμό εις πέρας τις απαιτήσεις του πελάτη. Καθώς το λογισμικό που μας ζητείται να αναπτύξουμε είναι απαιτητικό όσον αφορά το φόρτο εργασίας, κρίναμε απαραίτητη τη συμμετοχή 5 ατόμων στην ομάδα, προκειμένου η ανάθεση ευθυνών να είναι δικαιότερη και να έχουμε αποδοτικότερη υλοποίηση.

2 Αλλαγές

Για το έγγραφο **Team-plan-v1.0** πραγματοποιήθηκαν οι εξής τροποποιήσεις:

- Προσθήκη συμπερασμάτων από τη συνεργασία της ομάδας και κατανομή προσπάθειας.

Η προσθήκη σημειώνεται επίσης με [γαλάζιο χρώμα](#).

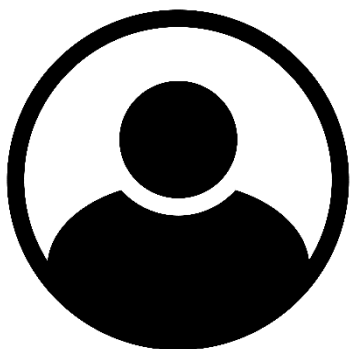
3 Μοντέλο – Αρμοδιότητες

Βασικό μας μέλημα στο οποίο δώσαμε ιδιαίτερη σημασία είναι η ομαδική συνεργασία για την επίλυση των ζητημάτων που προκύπτουν κατά τη διαδικασία ανάπτυξης του λογισμικού. Η βασική μας αρχή όσον αφορά την ανάπτυξη του έργου επιθυμούμε να βασίζεται σε μια εξελικτική προσέγγιση όπου κάθε εργασία θα υλοποιείται βηματικά μέχρις ότου να φτάσει στη τελική της μορφή. Επιπλέον, επιθυμούμε να υπάρχουν σαφείς ρόλοι τους οποίους θα αναλαμβάνει κάθε μέλος σε κάθε παραδοτέο και μάλιστα η ανάθεση αυτή να γίνεται εκ περιτροπής. Στόχος αυτού είναι κάθε μέλος να έχει τη δυνατότητα να αποκτήσει εμπειρία σε όλες τις διαφορετικές θέσεις που χρησιμοποιεί το μοντέλο εργασίας. Γι' αυτό το λόγο λοιπόν το μοντέλο που επιλέξαμε να εφαρμόσουμε είναι το **SCRUM**, το οποίο διαθέτει τους εξής ρόλους: τον **SCRUM Master**, τον **Product Owner** (προαιρετικός, τον οποίο και επιλέγουμε να μην υιοθετήσουμε) και το **Development Team**. Το μοντέλο αυτό μας επιτρέπει στην πράξη να εκτελέσουμε για κάθε παραδοτέο έναν sprint cycle, δηλαδή μια έκδοση του τελικού προϊόντος, η οποία θα αξιολογείται και θα τροποποιείται από όλα τα μέλη της ομάδας μέσα σε ένα συγκεκριμένο χρονικό διάστημα. Στη συνέχεια, η έκδοση αυτή θα παρουσιάζεται στον πελάτη και εφόσον εγκριθεί θα μεταβαίνουμε σε ένα επόμενο sprint cycle για την υλοποίηση του επόμενου παραδοτέου. Τέλος, κρίνουμε ότι το μοντέλο αυτό είναι δανικό για την ομάδα μας, καθώς επιδιώκουμε μια πιο δομημένη προσέγγιση μέσα σε συγκεκριμένα χρονικά περιθώρια.



SCRUM Master

Ο **SCRUM Master** αναλαμβάνει να οργανώνει καθημερινές συναντήσεις, να ελέγχει το backlog της δουλειάς που πρέπει να γίνει, να καταγράφει αποφάσεις, να μετράει πρόοδο και είναι ο μόνος που μπορεί να επικοινωνεί με πελάτη και διοίκηση.



Development Team

Είναι η ομάδα που αναλαμβάνει την πραγματοποίηση του έργου και την παράδοση των αποτελεσμάτων. Τα μέλη του **Development Team** εργάζονται μαζί για να προγραμματίσουν και να εκτελέσουν τις εργασίες που έχουν ανατεθεί σε κάθε sprint. Κατά την ολοκλήρωση κάθε sprint, το development team παραδίδει στο πελάτη λειτουργικά αποτελέσματα του έργου.

Για τις ανάγκες της συγκεκριμένης εργασίας κρίθηκε απαραίτητο να συμμετέχουν όλα τα μέλη της ομάδας στο Development Team, συμπεριλαμβανομένου και του **SCRUM Master** μαζί με τα υπόλοιπα καθήκοντά του. Πιο συγκεκριμένα, ο **SCRUM Master** θα εναλλάσσεται σε κάθε παραδοτέο και μάλιστα ανά τακτά χρονικά διαστήματα καθ' όλη τη διάρκεια υλοποίησής του. Στόχος μας είναι όλα τα μέλη της ομάδας να περάσουν από τη θέση του υπεύθυνου, ο οποίος καθορίζει όλο το πλάνο οργάνωσης και λειτουργίας της ομάδας σε κάθε συνάντηση. Η κατανομή των υπευθύνων αναλύεται εκτενέστερα στην **Ενότητα 4: Χρονοπρογραμματισμός**.

Την ίδια μεθοδολογία επιλέγουμε να ακολουθήσουμε και όσον αφορά την σύνταξη των τεχνικών αναφορών, στις οποίες θα διακρίνουμε τους εξής ρόλους:

- 1) **Author**: αναλαμβάνει τη συγγραφή του τεχνικού κειμένου.
- 2) **Co-author**: λειτουργεί επικουρικά βοηθώντας τον Author στη συγγραφή των κειμένων.
- 3) **Stylistics Editor**: αναλαμβάνει οτιδήποτε σχετικό με τη μορφοποίηση και τα γραφιστικά της αναφοράς, δηλαδή αναλαμβάνει την διαμόρφωση της τελικής αναφοράς που θα παραδοθεί.
- 4) **Reviewer**: είναι τα υπόλοιπα μέλη της ομάδας, τα οποία ελέγχουν την τελική αναφορά για τυχόν λάθη ή ανακρίβειες και προτείνουν διορθώσεις.

Οι ρόλοι αυτοί θα εναλλάσσονται κατόπιν σύσκεψης της ομάδας μετά την ολοκλήρωση του κάθε τεχνικού κειμένου και θα απεικονίζονται στην **Ενότητα 1: Σύνθεση Ομάδας** κάθε τεχνικού κειμένου, κάτω από τις φωτογραφίες των μελών μαζί με τα λοιπά στοιχεία τους.

4 Χρονοπρογραμματισμός

4.1 Συνεργασία και Οργάνωση

Αρχικά, πραγματοποιήσαμε μια αναγνωριστική συνάντηση, κατά την οποία προτάθηκαν διάφορες ιδέες για υλοποίηση, από τις οποίες μετά από ομόφωνη απόφαση καταλήξαμε στην εφαρμογή με όνομα **ReadEase**. Στη συνέχεια, καταστρώσαμε το πλάνο εργασίας που θα ακολουθήσουμε, σύμφωνα με το οποίο θα πραγματοποιούνται τακτικές συναντήσεις (ιδανικά δύο φορές την εβδομάδα και δια ζώσης) ανάμεσα σε όλα τα μέλη της ομάδας. Για έκτακτες περιπτώσεις όπου για κάποιο / κάποια μέλη δεν κατέστη δυνατή η δια ζώσης παρουσία, αποφασίστηκε η δυνατότητα συμμετοχής μέσω της ηλεκτρονικής πλατφόρμας **Zoom**, προκειμένου όλα τα μέλη να μπορούν να συμβαδίζουν. Οι ώρες συνάντησης κρίθηκε ότι πρέπει να καθοριστούν με κριτήριο την επίτευξη της μέγιστης αποδοτικότητας των μελών, δηλαδή κατά βάση πρωινές και μεσημεριανές ώρες.

Σε κάθε συνάντηση πραγματοποιούνται τα ακόλουθα:

- 1) Αξιολόγηση της δουλειάς που έχει ανατεθεί από την προηγούμενη συνέλευση,
- 2) Συζήτηση για βελτίωση της υλοποίησης και διόρθωση τυχόν σφαλμάτων,
- 3) Εκ νέου ισομοιρασμός καινούργιων αρμοδιοτήτων, καταγραφή τους και καθορισμός deadlines μέχρι την επόμενη συνάντηση.

ΣΗΜΕΙΩΣΗ

Ανάλογα με το βαθμό δυσκολίας και το πλήθος των παραδοτέων υπάρχει δυνατότητα συνεργασίας ανάμεσα σε 2 – 3 άτομα, προκειμένου ο επιπλέον φόρτος να καταμεριστεί καλύτερα (Programming in Pairs).

Τέλος, έχει αποφασιστεί οι αλλαγές καθώς και οι προσθήκες να αποθηκεύονται σε κοινόχρηστο αποθετήριο, έτσι ώστε όλα τα μέλη να έχουν πρόσβαση στα παραδοτέα και το κώδικα ανά πάσα στιγμή για τυχόν αλλαγές που μπορεί να χρειαστεί να πραγματοποιηθούν εκτός από τα πλαίσια της συνέλευσης. Επιπρόσθετα, στο αποθετήριο καταγράφονται οι αλλαγές που πραγματοποιήθηκαν, σε ποια σημεία του κώδικα έγιναν, από ποιο μέλος καθώς και ποια χρονική στιγμή. Αυτή η δυνατότητα μας παρέχει την ευκολία παρακολούθησης της εξέλιξης του έργου μας καθ' όλη τη διάρκεια του εξαμήνου. Περισσότερες λεπτομέρειες για το αποθετήριο υπάρχουν στην **Ενότητα 6: Εργαλεία** του παρόντος εγγράφου.

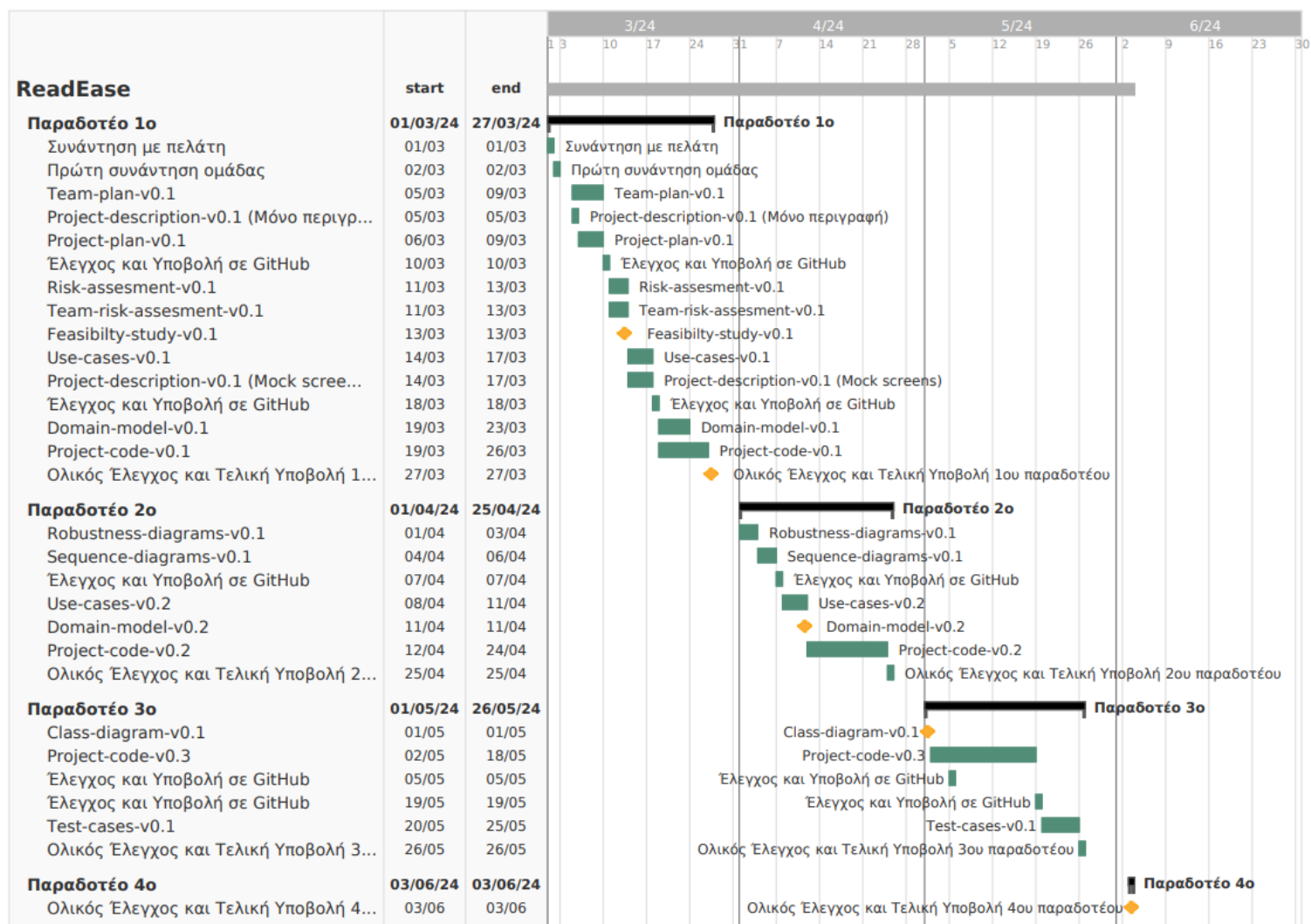
Οι συγκεκριμένες ενέργειες που αναφέρονται παραπάνω πραγματοποιούνται σε κάθε συνάντηση από τον υπεύθυνο της ομάδας. Ο υπεύθυνος της ομάδας έχει συμφωνηθεί να αλλάζει μετά την ολοκλήρωση ενός milestone, δηλαδή ενός σημείου το οποίο κρίνουμε ότι είναι υψίστης σημασίας για την ολοκλήρωση του έργου. Μοναδική εξαίρεση αποτελεί η διαδικασία υποβολής της τεχνικής αναφοράς στο **GitHub**, την οποία αναλαμβάνει να τη κάνει ο Author σύμφωνα με το χρονοδιάγραμμα που παρέχεται, κάνοντάς την έτσι διαθέσιμη σε όλα τα μέλη για έλεγχο και τροποποιήσεις.

Στη διπλανή σελίδα φαίνεται ένας ενδεικτικός πίνακας με όλα τα milestones, την εκτίμηση της χρονικής στιγμής υλοποίησής τους, καθώς και το μέλος της ομάδας που έχει αναλάβει το ρόλο του **SCRUM Master** καθ' όλη τη διάρκειά του. Τα σημεία των milestones εμφανίζονται επιπλέον και στο Gantt Chart όπου παρουσιάζουμε όλο τον χρονοπρογραμματισμό του έργου μας, στην **Υποενότητα 4.2: Gantt Chart**.

Milestone	Date	SCRUM Master
Feasibility-study-v0.1	01.03.2024 – 13.03.2024	Χρυσανγή Πατέλη
Ολικός Έλεγχος και Τελική Υποβολή 1ου παραδοτέου	14.03.2024 – 27.03.2024	Γεώργιος Καραγιάννης
Domain-model-v0.2	01.04.2024 – 11.04.2024	Μαρία Κουρή
Class-diagram-v0.1	12.04.2024 – 01.05.2024	Μηλιτιάδης Μαντές
Ολικός Έλεγχος και Τελική Υποβολή 4ου παραδοτέου	02.05.2024 – 03.06.2024	Ευαγγελία Κολάγκη

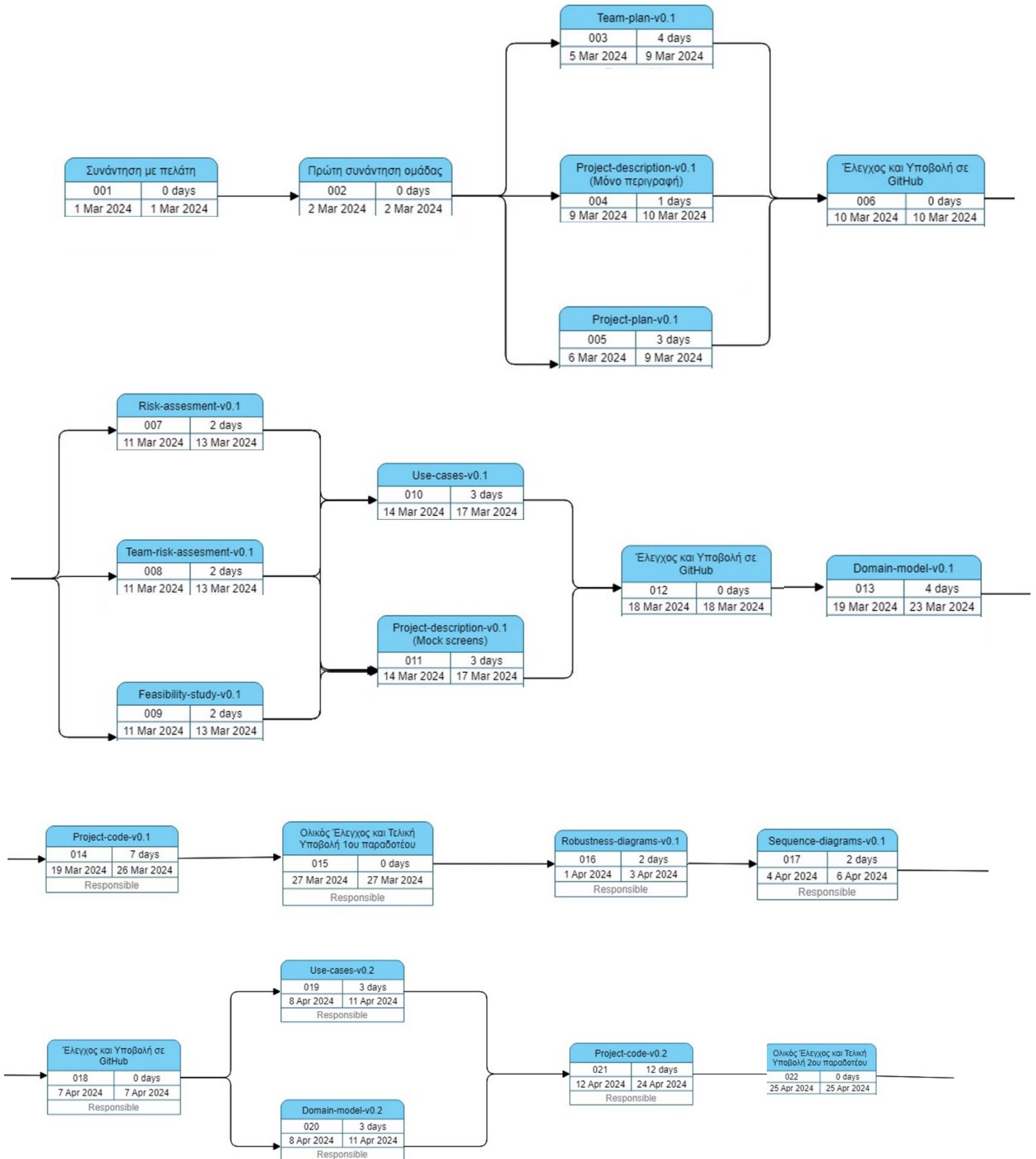
Πίνακας 1: Κατανομή των SCRUM Masters

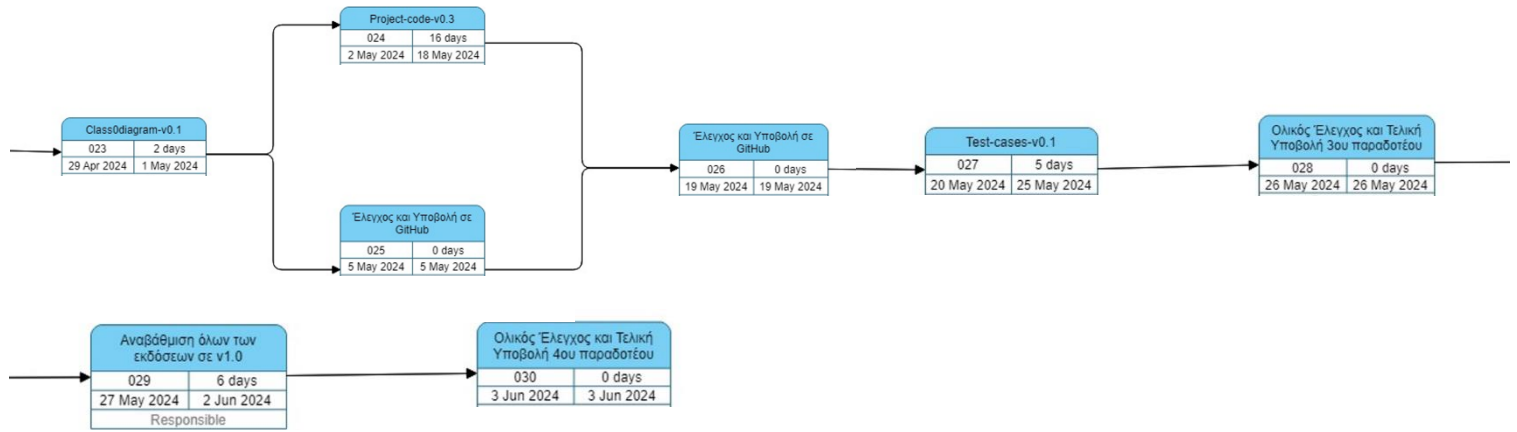
4.2 Gantt Chart



Πίνακας 2: Gantt Chart

4.3 Pert Chart





Πίνακας 3: Pert Chart

5 Κατανομή Προσπάθειας και Συμπεράσματα

Κάνοντας μια συνολική ανασκόπηση της εργασίας καταλήξαμε ότι δεν υπήρξαν σημαντικές αποκλίσεις από τις αρχικές μας προσδοκίες, καθώς ακολουθήσαμε με ακρίβεια τις μεθόδους εργασίας που είχαμε επιλέξει από την αρχή και προσπαθήσαμε να ανταπεξέλθουμε όσο καλύτερα γίνεται στο χρονοδιάγραμμα που ορίσαμε. Γενικά, δεν αντιμετωπίσαμε μεγάλο αριθμό προβλημάτων και όλοι μείναμε ικανοποιημένοι από τη συνεργασία μας παρά το βεβαρημένο πρόγραμμα όλων των μελών της ομάδας λόγω ακαδημαϊκών υποχρεώσεων και φόρτου εργασίας. Συνεπώς, όλα τα μέλη της ομάδας έφεραν εις πέρας τις αρμοδιότητες που τους είχαν ανατεθεί και δεν υπήρξε κανένα απρόοπτο κατά τη διάρκεια αυτής της συνεργασίας.

Όσον αφορά τη κατανομή της προσπάθειας, όλα τα μέλη της ομάδας συμφωνούν **ομόφωνα** ότι υπήρξε ισοδύναμη συμβολή όλων των μελών στην εργασία και άρα καταλήγουμε στον παρακάτω ισόποσο διαμερισμό:

- **E₁: Γεώργιος Καραγιάννης: 0.20**
- **E₂: Ευαγγελία Κολάγκη: 0.20**
- **E₃: Μαρία Κουρή: 0.20**
- **E₄: Μηλιτιάδης Μαντές: 0.20**
- **E₅: Χρυσauγή Πατέλη: 0.20**

6 Εργαλεία



MS Word

Θα χρησιμοποιηθεί για την σύνταξη και την επεξεργασία όλων των τεχνικών κειμένων.



GitHub

Θα χρησιμοποιηθεί για την οργάνωση, αποθήκευση και τροποποίηση του κώδικα, το ανέβασμα των τεχνικών αναφορών και τη παρακολούθηση της εξέλιξης του έργου και του ιστορικού υποβολών.



teamgantt

Θα χρησιμοποιηθεί για την εξαγωγή του Gantt Chart.



Visual Paradigm

Θα χρησιμοποιηθεί για την εξαγωγή του Pert Chart.



Canva

Θα χρησιμοποιηθεί για τον σχεδιασμό του λογότυπου της εφαρμογής και των mock screens που θα βλέπει ο χρήστης στην οθόνη του.



Git

Θα χρησιμοποιηθεί για την αποθήκευση και τροποποίηση κώδικα και τη μεταφορά του στο GitHub.



PyCharm

Η αντικειμενοστρεφής γλώσσα που επιλέγεται για την υλοποίηση του έργου είναι αυτή της Python.



MySQL

Χρησιμοποιείται για τη κατασκευή της βάσης δεδομένων.