

# Team-plan-v0.2

*Κωδικός: Team-plan*

*Έκδοση: v0.2*

## Movfast



## Μέλη της ομάδας

---

Ονοματεπώνυμο	Αρ. Μητρώου	Έτος Σπουδών
Νικόλαος Ανδριανόπουλος	1084637	4 <sup>ο</sup>
Παναγιώτης Καλοζούμης	1084560	4 <sup>ο</sup>
Βασίλειος – Μάριος Κουρτάκης	1090061	4 <sup>ο</sup>
Δημήτριος Στασινός	1084643	4 <sup>ο</sup>

## Συμμετέχοντες στο παρόν κείμενο

---

- Παναγιώτης Καλοζούμης (Editor)
- Νικόλαος Ανδριανόπουλος (Peer Reviewer)

## Αλλαγές σε σχέση με την προηγούμενη έκδοση

---

1. Στην Εκτίμηση Εργαλείων, αλλάξαμε το εργαλείο που χρησιμοποιούμε για την παραγωγή του διαγράμματος κλάσεων από Microsoft Visio σε IntelliJ
2. Αναλύθηκε περισσότερο η μέθοδος εργασίας που θα ακολουθήσουμε:
  - Εξηγήσαμε ότι η Scrum είναι μια Agile μέθοδος
  - Προσθέσαμε μια αναφορά στο product backlog, κάνοντας έτσι μια διαφοροποίηση μεταξύ αυτού και του project backlog (sprint backlog)

- Αναλύσαμε περισσότερο τι περιλαμβάνει ένα scrum sprint, διακρίνοντας τις 4 φάσεις του και εξηγώντας συνοπτικά τι περιλαμβάνει κάθε μια απ' αυτές

## Μέθοδος εργασίας

---

Η μέθοδος εργασίας που θα ακολουθήσουμε για την ανάπτυξη του έργου είναι η Scrum. Η Scrum εντάσσεται στις Agile μεθόδους ανάπτυξης λογισμικού, βασίζεται δηλαδή στην σταδιακή ανάπτυξη και παράδοση του λογισμικού και όχι στον λεπτομερή σχεδιασμό του συστήματος από την αρχή. Αυτές οι μέθοδοι λειτουργούν καλά για μεσαίου μεγέθους έργα, όπου οι απαιτήσεις ενδεχομένως αλλάζουν με σημαντικό ρυθμό και στόχος είναι η γρήγορη αποστολή νέων χαρακτηριστικών στους πελάτες για τη λήψη αναπληροφόρησης. Η μέθοδος Scrum χωρίζεται σε τρεις φάσεις: φάση σχεδίασης, sprint cycles και τελική φάση.

Στην αρχική φάση θα καθορίσουμε τους στόχους του έργου και θα δοθεί ιδιαίτερη έμφαση στη σχεδίαση της βασικής αρχιτεκτονικής της εφαρμογής μας, ώστε να μπορούμε να προχωρήσουμε όσο πιο ομαλά γίνεται στον προγραμματισμό της στη συνέχεια. Στη φάση αυτή συντίθεται το product backlog, μια λίστα από όλες τις εργασίες που πρέπει να ολοκληρωθούν κατά την ανάπτυξη.

Μόλις ολοκληρωθεί η φάση αυτή, μπαίνουμε στη διαδικασία ανάπτυξης του λογισμικού. Η φάση αυτή είναι επαναληπτική, καθώς αποτελείται από μια σειρά από scrum sprints. Κάθε sprint αναμένουμε να διαρκεί 2-3 βδομάδες και χωρίζεται περαιτέρω σε 4 στάδια: αξιολόγηση, επιλογή, ανάπτυξη και έλεγχος. Η πρώτη φάση αφορά την τροποποίηση του product backlog. Εδώ αξιολογούμε τα περιεχόμενα του product backlog, αναθέτοντας προτεραιότητες σ' αυτά και εισάγοντας νέες απαιτήσεις, αν το κρίνουμε απαραίτητο. Εδώ να σημειωθεί ότι, εφόσον δεν υπάρχει πελάτης στην περίπτωσή μας, προσθέτουμε νέες απαιτήσεις στο backlog, σε αντίθεση με ένα πραγματικό σενάριο όπου υπάρχει συνεργασία με τον πελάτη. Έπειτα επιλέγουμε ένα υποσύνολό του product backlog, το project/sprint backlog, το οποίο περιέχει τα χαρακτηριστικά της εφαρμογής που θα υλοποιηθούν πλήρως κατά το τρέχον sprint.. Αφού αποφασίσουμε τι χαρακτηριστικά της εφαρμογής θα αναπτύξουμε, συνεχίζουμε με τον προγραμματισμό.

Η διαδικασία της ανάπτυξης, παράλληλα με τη συγγραφή κώδικα, περιλαμβάνει και σύντομες συναντήσεις όλων των μελών της ομάδας. Κύρια ζητήματα των συναντήσεων αυτών θα είναι η αναφορά της προόδου των μελών πάνω στις εργασίες που τους έχουν ανατεθεί, η ανάθεση νέων εργασιών από τη λίστα αν κάποιες έχουν ήδη ολοκληρωθεί και η αναφορά τυχόν προβλημάτων που προέκυψαν στην ανάπτυξη του λογισμικού. Οι

συναντήσεις αυτές οργανώνονται από τον Scrum Master της ομάδας. Σε σχέση με την κλασσική Scrum, η δική μας μέθοδος εργασίας διαφοροποιείται ελαφρώς εδώ, ώστε να είναι προσαρμοσμένη στις δικές μας συνθήκες:

- Το Daily Scrum αντικαθίσταται από εβδομαδιαίες συναντήσεις, συνήθως στο τέλος της εβδομάδας. Ο αριθμός των συναντήσεων δεν είναι δεσμευτικός και μπορεί ενδεχομένως να αυξάνεται κατ' απαίτηση, ανάλογα με την πολυπλοκότητα των εργασιών που πρέπει να ολοκληρωθούν και αν το κρίνει απαραίτητο ο Scrum Master.
- Λόγω της χαμηλότερης συχνότητας των συναντήσεων, ενδέχεται να είναι λίγο μεγαλύτερης διάρκειας σε σχέση με μια κλασσική συνάντηση Scrum. Εξακολουθούν, ωστόσο, να διατηρούν τη λογική των σύντομων συναντήσεων.
- Οι συναντήσεις μας θα γίνονται κυρίως εξ' αποστάσεως.

Οι παραπάνω αλλαγές έγιναν λαμβάνοντας υπ' όψη και τις υπόλοιπες υποχρεώσεις των μελών της ομάδας, καθώς οι καθημερινές συναντήσεις στα πλαίσια του εξαμήνου δεν είναι εφικτές. Επίσης θεωρήσαμε ότι οι συχνές δια ζώσης συναντήσεις θα αποσπούσαν πολύτιμο χρόνο από τη διαδικασία της ανάπτυξης, άρα προτιμήθηκε να γίνονται εξ' αποστάσεως.

Στην τελική φάση της εργασίας με Scrum, όπου το προγραμματιστικό έργο θα έχει ολοκληρωθεί στο σύνολό του, σκοπεύουμε να καταγράψουμε τις εμπειρίες μας για την εργασία μας ως ομάδα και την απόδοση της μεθόδου Scrum, αναφέροντας τι ενδεχομένως θα αλλάζαμε στον τρόπο που χειριστήκαμε το έργο.

Κλείνοντας, αναφέρουμε τους ρόλους κατά Scrum που έχουμε αναθέσει σε κάθε μέλος της ομάδας. Επιλέξαμε οι ρόλοι αυτοί να μείνουν σταθεροί καθ' όλη τη διάρκεια του project:

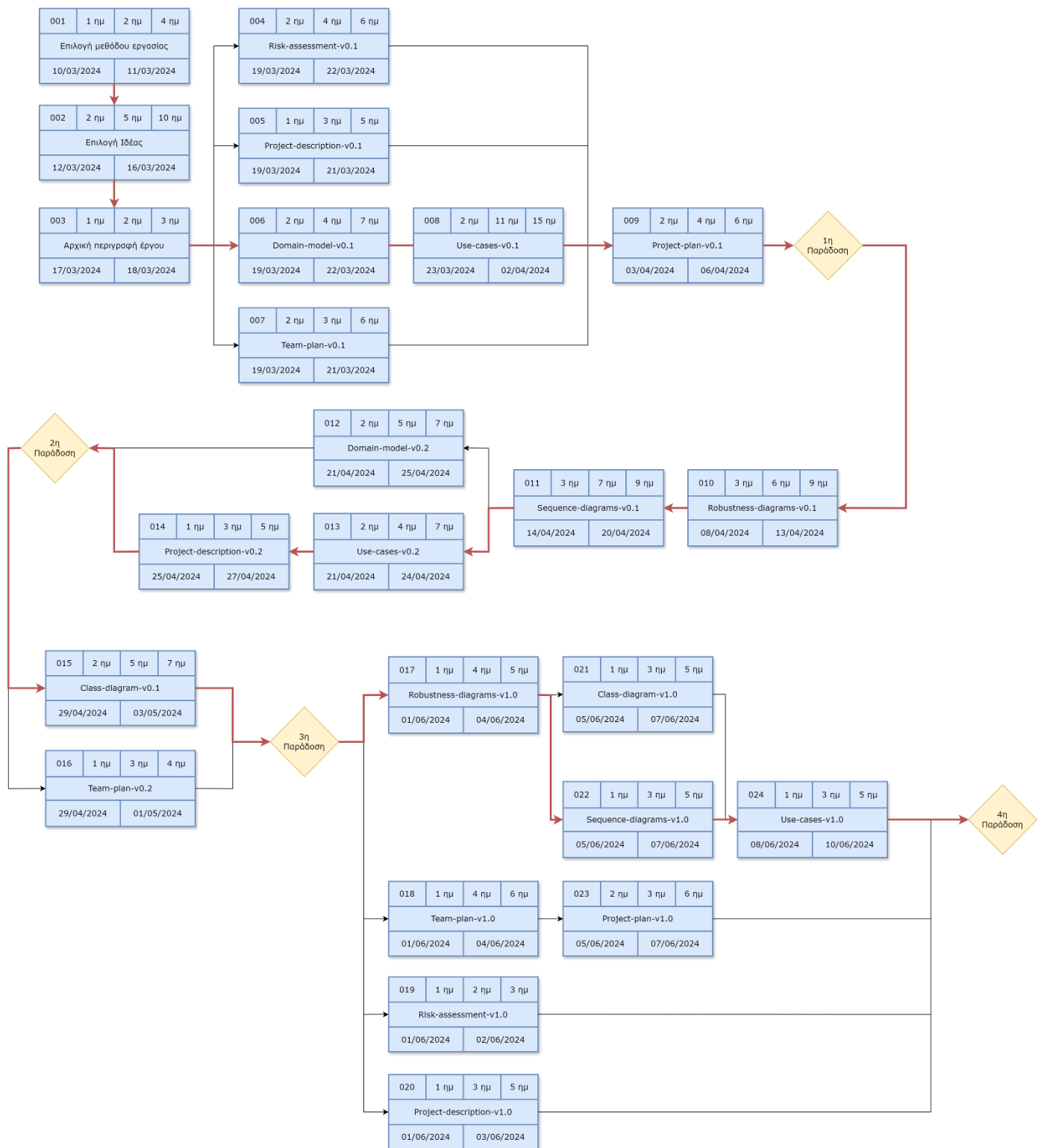
- **Product Owner – Νικόλαος Ανδριανόπουλος:** Διαχειρίζεται το backlog, επιλέγοντας τα περιεχόμενά του, αξιολογώντας τα και αναθέτοντας προτεραιότητες σε αυτά στην αρχή κάθε sprint.
- **Scrum Master – Παναγιώτης Καλοζούμης:** Καθοδηγεί και συντονίζει την ομάδα, οργανώνει συναντήσεις και παρακολουθεί την πρόοδο της ομάδας ως προς το backlog. Φροντίζει επίσης ότι όλα τα μέλη ακολουθούν τις πρακτικές που ορίζει η μέθοδος Scrum. Θεωρητικά σε ένα πραγματικό έργο το μέλος αυτό θα ήταν υπεύθυνο και για την επικοινωνία με τον πελάτη κατά τη διάρκεια του sprint.

- **Team Members – Όλη η ομάδα:** Η ομάδα προγραμματισμού. Είναι τα μέλη που έχουν όλες τις απαιτούμενες γνώσεις προγραμματισμού για τη διεκπεραίωση του έργου. Τα μέλη έχουν μεγάλο βαθμό αυτονομίας και πρέπει να μπορούν να αυτό-οργανώνονται.

Να σημειωθεί ότι στα πλαίσια του project, και καθώς εργαζόμαστε μόνο σε μια μικρογραφία ενός πραγματικού έργου, είναι εφικτό, και ιδιαίτερα σημαντικό, και τα τέσσερα μέλη της ομάδας να συμμετέχουν ενεργά στη συγγραφή κώδικα. Επομένως, στην περίπτωση μας, όλα τα μέλη θεωρούνται και μέλη του Scrum Team.

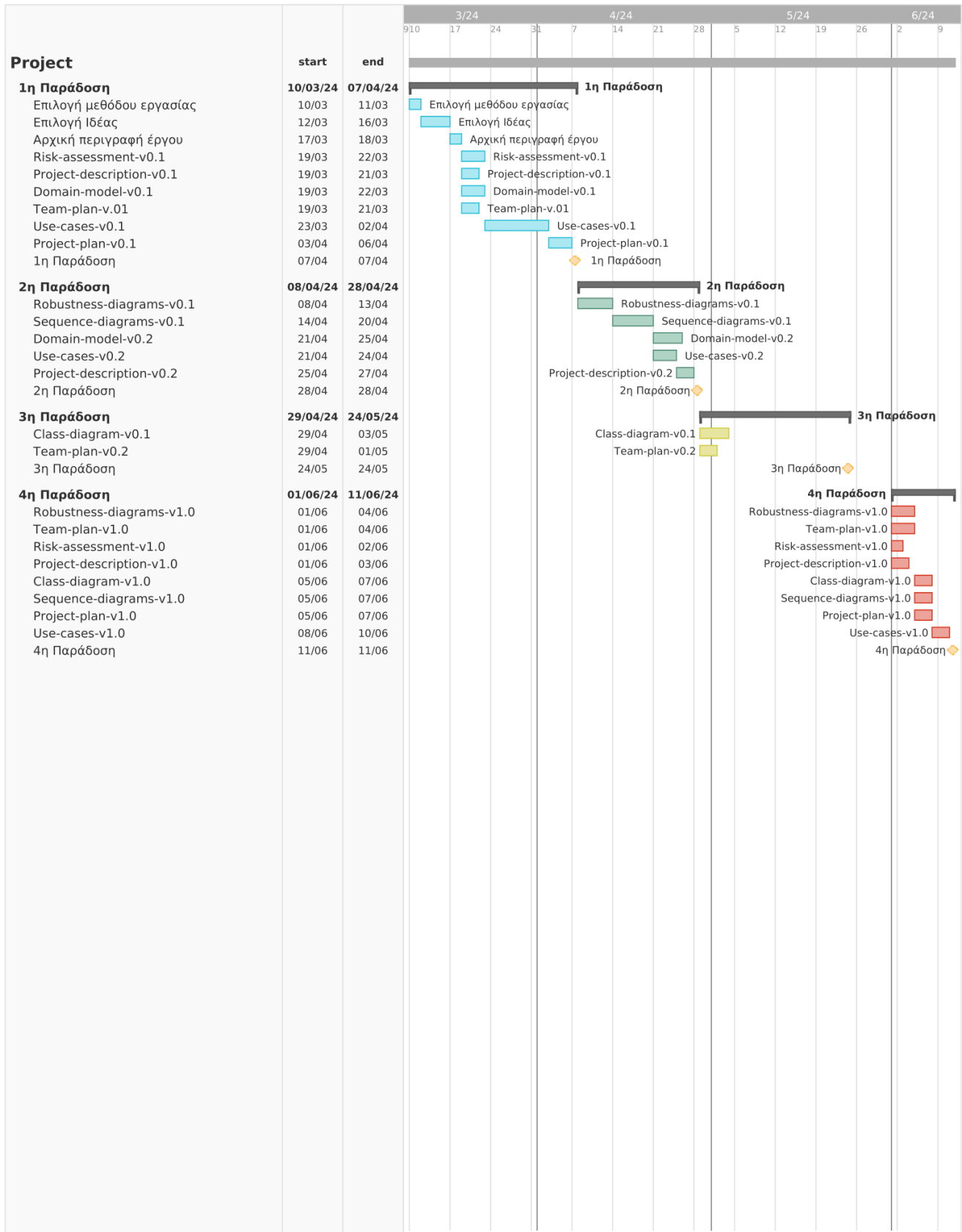
## Χρονοπρογραμματισμός project

Παρακάτω παραθέτουμε τον αρχικό χρονοπρογραμματισμό των τυπικών υποέργων. Σαν τυπικά υποέργα έχουμε υποθέσει τα τεχνικά κείμενα κάθε κύκλου παράδοσης. Προσπαθήσαμε να εκτιμήσουμε τον χρόνο ολοκλήρωσης κάθε υποέργου όσο πιο ρεαλιστικά γίνεται, λαμβάνοντας υπ' όψη και υποχρεώσεις των μελών της ομάδας ως προς τα υπόλοιπα μαθήματα του εξαμήνου. Έχουμε, επομένως, αφήσει ένα μικρό χρονικό περιθώριο επιπλέον για την ολοκλήρωση κάθε τυπικού υποέργου. Αναμένουμε, ωστόσο, καθώς αποκτάμε εμπειρία στον τρόπο εργασίας μας ως ομάδα, ο χρονοπρογραμματισμός αυτός να τροποποιηθεί ελαφρώς στους επόμενους κύκλους παράδοσης.

**Διάγραμμα Pert:**

Εικόνα 1 Διάγραμμα Pert παραδοτέων

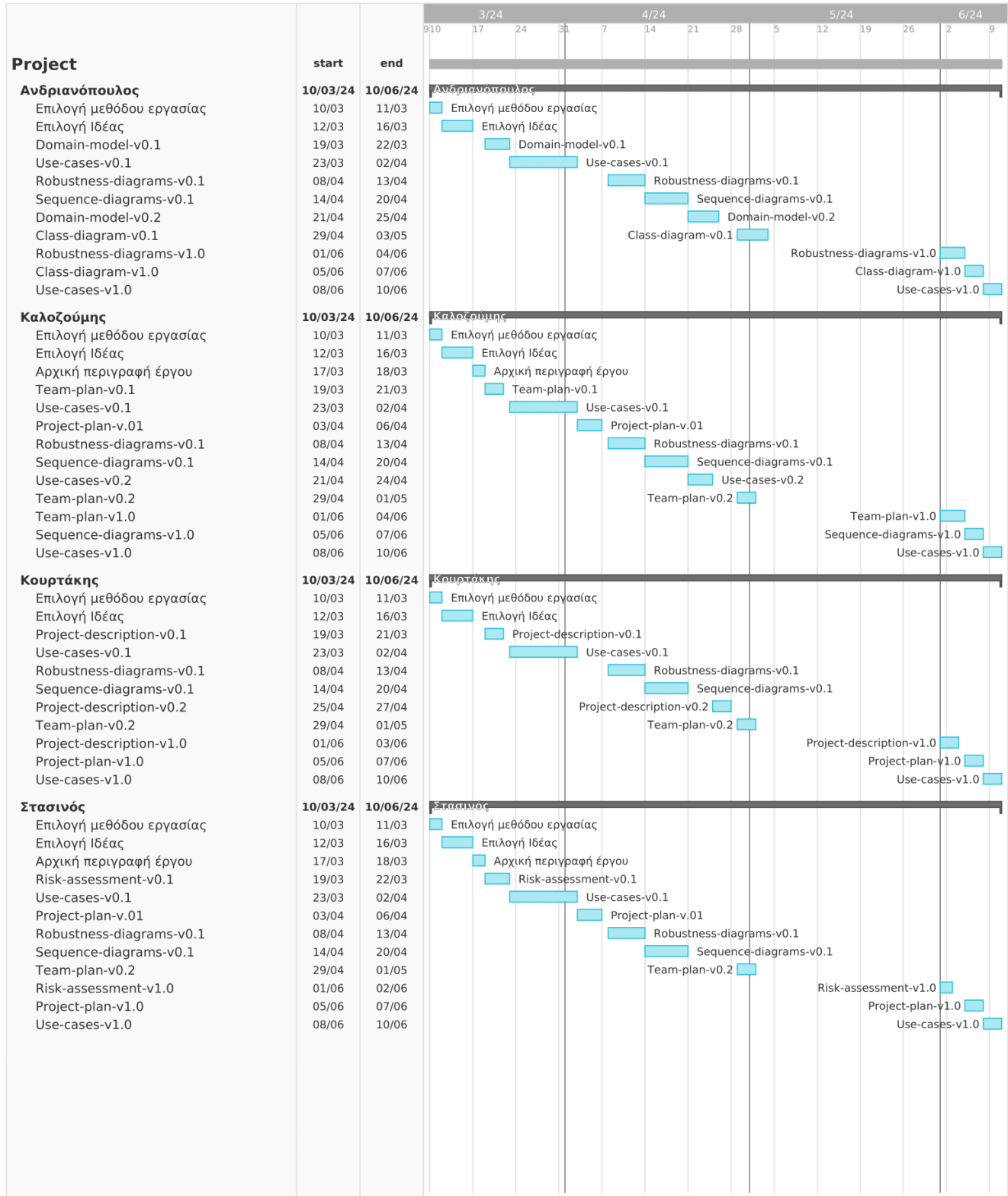
## Διάγραμμα Gantt:



Εικόνα 2 Διάγραμμα Gantt χρονοπρογραμματισμού παραδοτέων

## Ανάθεση έργου σε ανθρώπινο δυναμικό

Παρακάτω παραθέτουμε τον διάγραμμα Gantt ανάθεσης έργου, σύμφωνα με τον αρχικό χρονοπρογραμματισμό που δείξαμε παραπάνω:



Εικόνα 3 Διάγραμμα Gantt ανάθεσης παραδοτέων



## Εκτίμηση εργαλείων

---

Στη συνέχεια παραθέτουμε τα εργαλεία που σκοπεύουμε να χρησιμοποιήσουμε κατά τη διάρκεια του project.

Σχετικά με τα τεχνικά κείμενα:

- **Συγγραφή τεχνικών κειμένων:** Microsoft Word
- **Δημιουργία Pert Chart:** draw.io
- **Δημιουργία Gantt Chart:** TeamGantt
- **Δημιουργία mockup screens:** Figma
- **Δημιουργία Use Case Diagram:** draw.io
- **Δημιουργία Robustness Diagram:** draw.io
- **Δημιουργία Sequence Diagram:** Microsoft Visio
- **Δημιουργία Domain Model:** Microsoft Visio
- **Δημιουργία Class Diagram:** IntelliJ

Σχετικά με τη συγγραφή κώδικα:

- **Γλώσσα προγραμματισμού:** Java (Android)
- **IDE:** Android Studio
- **Version Control:** GitHub

## Εργαλεία που χρησιμοποιήθηκαν στο παρόν κείμενο

---

Τα εργαλεία που χρησιμοποιήθηκαν για τη συγγραφή του Team-plan είναι:

- **Συγγραφή:** Microsoft Word
- **Δημιουργία Pert Chart:** draw.io
- **Δημιουργία Gantt Chart:** TeamGantt