

ΤΕΧΝΟΛΟΓΙΑ ΛΟΓΙΣΜΙΚΟΥ

Εαρινό Εξάμηνο 2022-2023

<u>Ονοματεπώνυμο</u>	<u>A.M.</u>	<u>email</u>
Στεφανόπουλος Αλέξιος	1041895	ceid6219@upnet.gr
Θοδωρής Τσουκαλαδέλης	1070909	up1070909@upnet.gr
Σάαντ Γεώργιος	1041870	ceid6193@upnet.gr
Γιώργος Παπανδρέου	1067506	up1067506@upnet.gr

TEAM PLAN v0.1

Table of Contents

[1. Team Plan \(v0.1\)](#)

[1.1. Διαδικαστικά ομάδας](#)

[1.1.1. SCRUM](#)

[1.2. Χρονοπρογραμματισμός](#)

[1.2.1. Gantt](#)

[1.2.2. Pert](#)

[1.3. Εργαλεία](#)

[1.3.1. Software Development](#)

[1.3.2. Management](#)

1. Team Plan

1.1. Διαδικαστικά ομάδας

Η ομάδα αποτελείται από 4 άτομα. Όπως φαίνεται στον παρακάτω πίνακα. Στον πίνακα φαίνεται επίσης ο ρόλος του κάθε μέλους στο μοντέλο SCRUM που χρησιμοποιούμε (οι πληροφορίες του οποίου βρίσκονται στο 1.1.1)

Όνοματεπώνυμο	ΑΜ	Ρόλος (SCRUM)
Στεφανόπουλος Αλέξιος	1041895	SCRUM Master, Developer
Τσουκαλαδέλης Θεόδωρος	1070909	Product Owner, Developer
Σάαντ Γεώργιος	1041870	Developer
Παπανδρέου Γεώργιος	1067506	Developer

1.1.1. SCRUM

Ο τρόπος που θα δουλέψει η ομάδα είναι με SCRUM. Το SCRUM αποτελείται από διάφορους ρόλους, ο καθένας εκ των οποίων έχει και μια συγκεκριμένη αρμοδιότητα. Στη δική μας ομάδα, οι ρόλοι θα έχουν ως εξής.

1. Product Owner

Στη περίπτωση της εργασίας δεν υπάρχουν stakeholders με τους οποίους θα μπορούσε ένας Product Owner να βρεί κοινά σημεία μεταξύ αυτών και της ίδιας της ομάδας. Για αυτό το λόγο, ο Product Owner θα έχει κυρίως την ευθύνη για τον καθορισμό σωστών προτεραιοτήτων στο έργο, και τη καθοδήγηση της ομάδας σε σημεία όπως τις τακτικές με τις οποίες θα μπορούσαμε να κάνουμε market ένα προϊόν.

2. Scrum Master

Ο Scrum Master θα καθοδηγεί την ομάδα κυρίως γύρω από τα milestones του κάθε sprint,

και θα οργανώνει επίσης όλες τις κλήσεις στις οποίες θα παρευρίσκονται όλα τα μέλη της ομάδας για συζήτηση πάνω στα παραδοτέα του τωρινού sprint, στη δουλειά που έχει γίνει ή και γίνεται, καθώς και τυχών βελτιώσεις.

3. Developers

Η δουλειά αυτού του ρόλου αφορά οποιαδήποτε εργασία επάνω σε κάποιο παραδοτέο. Εφόσον κάθε μέλος της ομάδας μας έχει δουλέψει και θα δουλέψει στα παραδοτέα, τότε θα είναι όλοι και developers.

Αξιζει επίσης να σημειωθεί πως δεν είναι δυνατό να βρίσκεται η ομάδα κάθε μέρα για SCRUM sessions, οπότε γίνεται επικοινωνία μεταξύ των μελών σε ένα μεγάλο session κάθε εβδομάδα στο οποίο γίνεται συζήτηση των ζητημάτων της εφαρμογής. Στο ίδιο session δημιουργούνται δύο υπο-ομάδες, δύο ατόμων η κάθε μια, που θα αναλαμβάνουν διαφορετικά παραδοτέα. Τα μέλη αυτών των ομάδων επικοινωνούν συχνότερα μεταξύ, ενώ γίνεται επικοινωνία με τον Scrum Master κάθε φορά που υπάρχουν σημαντικές αλλαγές/βελτιώσεις σε ένα παραδοτέο ώστε να μπορεί αυτός να συντονίσει τις ομάδες μεταξύ τους.

1.2. Παραδοτέα

Τα παραδοτέα είναι χωρισμένα σε milestones και στα ίδια τα τεχνικά κείμενα. Παρακάτω φαίνεται η λίστα τους, καθώς και μια συνοπτική περιγραφή του καθενός.

- D1 (Milestone) – Συγγραφή όλων των κειμένων που συμπεριλαμβάνονται σε αυτό.

- |- Project Description (Τεχνικό Κείμενο) – Σύνοψη της λειτουργία και των δυνατοτήτων της εφαρμογής
- |- Team Plan (Τεχνικό Κείμενο) – Περιγραφή οργάνωσης ομάδας project και τρόπος εργασίας
- |- Project Plan (Τεχνικό Κείμενο) – Χρονοπρογραμματισμός ανάπτυξης της εφαρμογής
- |- Risk Assessment (Τεχνικό Κείμενο) – Περιγραφή προβλημάτων που θα μπορούσε να συναντήσει το έργο, και λύσεις σε αυτά

- D2 (Milestone) – Συγγραφή όλων των κειμένων που συμπεριλαμβάνονται σε αυτό.

- |- Use Case (Τεχνικό Κείμενο) – Περιγραφή λειτουργιών και σημείων χρήσης της εφαρμογής
- |- Domain Model (Τεχνικό Κείμενο) – Περιγραφή ιεραρχίας οντότητων πεδίου εφαρμογής

- D3 (Milestone) - Συγγραφή όλων των κειμένων που συμπεριλαμβάνονται σε αυτό.

- |- Robustness Diagrams – Δημιουργία διαγράμματος ευρωστίας για κάθε use-case

- D4 (Milestone) - Συγγραφή όλων των κειμένων που συμπεριλαμβάνονται σε αυτό.

- |- Sequence Diagrams – Δημιουργία διαγράμματος ακολουθίας για κάθε use-case

- D5 (Milestone) - Συγγραφή όλων των κειμένων που συμπεριλαμβάνονται σε αυτό.

- |- Class Diagram – Δημιουργία διαγράμματος κλάσεων εφαρμογής
- |- Project Code – Codebase εφαρμογής

- D6 (Milestone) - Τελικές εκδόσεις σε όλα τα παραπάνω παραδοτέα

1.3. Χρονοπρογραμματισμός

Η ιεραρχία των τεχνικών κειμένων ως προς τις προθεσμίες τους και τη σειρά ολοκλήρωσης τους φαίνεται στον πίνακα παρακάτω. Διευκρινίζονται τα εξής:

- Στη στήλη 'Ανάθεση' αναφέρονται τα αρχικά των μελών της ομάδας τα οποία έχουν αναλάβει το συγκεκριμένο παραδοτέο.
- Στη στήλη 'Τύπος' αναφέρεται ο τύπος του παραδοτέου, έχει χρησιμοποιηθεί '**Μ**' για ένδειξη **Milestone**, και '**TK**' για ένδειξη **Τεχνικού Κειμένου**.
- Τα υποπαραδοτέα του κάθε Milestone κληρονομούν και τα παραδοτέα στα οποία βασίζεται αυτό. Με τη χρήση του αριθμού ενός Milestone στη στήλη 'Βασίζεται σε' (π.χ. 1.1, 1.4, κτλ), εννοείται η ολοκλήρωση όλων των υποπαραδοτέων αυτού ως προϋπόθεση.

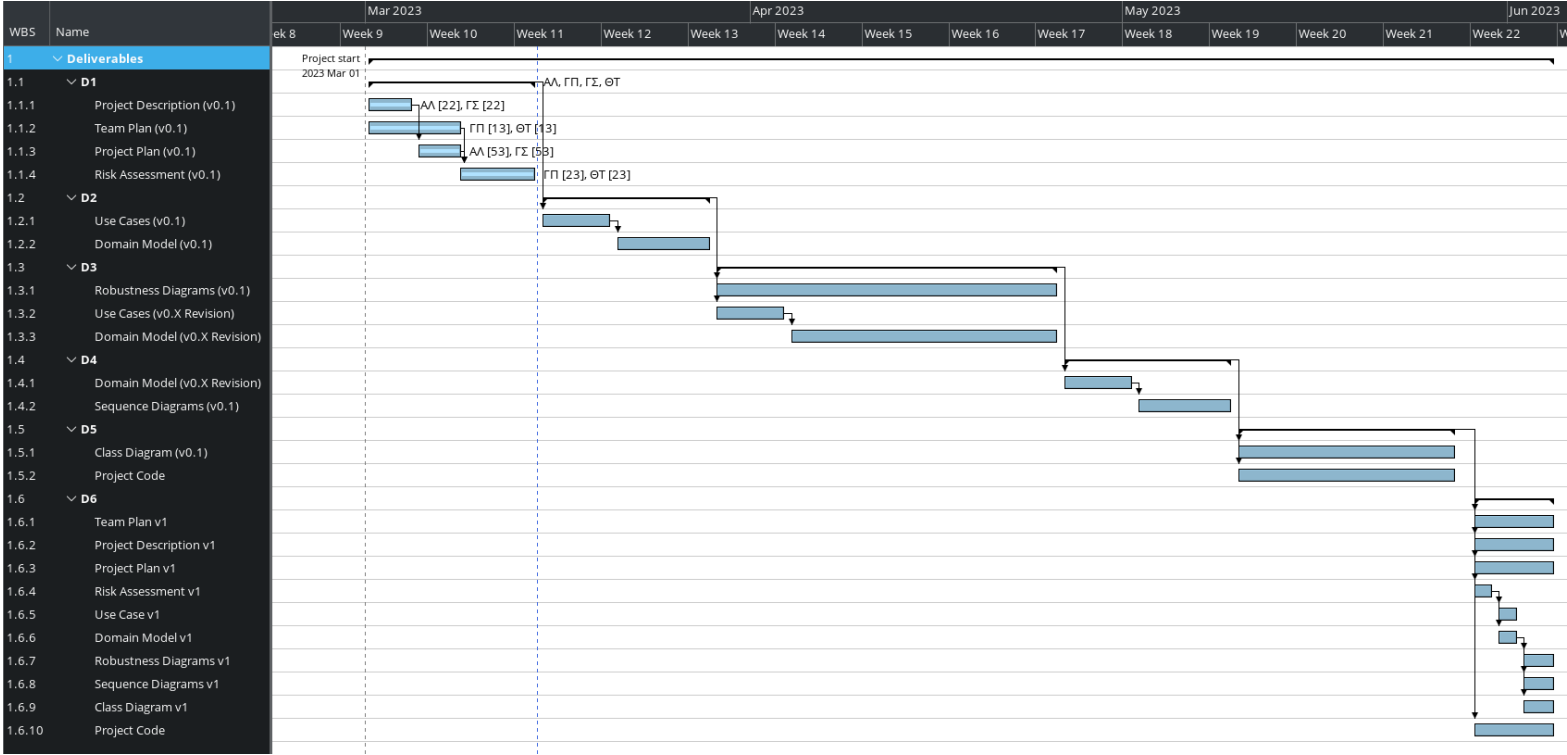
#	Τύπος	Παραδοτέο	Βασίζεται Σε	Έναρξη (Το αργότερο)	Προβλεπόμενη Ημερομηνία Ολοκλήρωσης	Τελική Ημερομηνία Ολοκλήρωσης	Εκτιμώμενη Διάρκεια (Ωρες)	Ανάθεση
1.1.1	TK	Project-description-v0.1	-	1/3	2/3	4/3	6-7	ΑΛ, ΓΣ
1.1.2	TK	Team-plan-v0.1	-	1/3	6/3	8/3	5-6	ΓΠ, ΘΤ
1.1.3	TK	Project-plan-v0.1	1.1.1	5/3	6/3	8/3	4-5	ΑΛ, ΓΣ
1.1.4	TK	Risk-assessment-v0.1	1.1.2, 1.1.3	9/3	14/3	14/3	7-8	ΓΠ, ΘΤ
1.1	M	D1	-	-	14/3	14/3	22-26	ΑΛ, ΓΠ, ΓΣ, ΘΤ
1.2.1	TK	Use-case-v0.1	1.1	15/3	18/3	20/3	8-9	
1.2.2	TK	Domain-model-v0.1	1.2.1	21/3	26/3	28/3	6-7	

1.2	M	D2	1.1	-	26/3	28/3	14-16	ΑΛ, ΓΠ, ΓΣ, ΘΤ
1.3.1	TK	Robustness- diagrams- v0.1	1.2	29/3	7/4	25/4	7-8	
1.3.2	TK	Use-cases- v0.X (Αναθεώρησ η)	1.2	29/3	1/4	3/4	7-8	
1.3.3	TK	Domain- model-v0.X (Αναθεώρησ η)	1.3.2	4/4	7/4	25/4	6-7	
1.3	M	D3	1.1, 1.2	-	7/4	25/4	20-23	ΑΛ, ΓΠ, ΓΣ, ΘΤ
1.4.1	TK	Domain- model-v0.X (Αναθεώρησ η)	1.3	26/4	1/5	1/5	7-8	
1.4.2	TK	Sequence- diagrams- v0.1	1.4.2	1/5	4/5	9/5	5-6	
1.4	M	D4	1.1, 1.2, 1.3	-	4/5	9/5	12-14	ΑΛ, ΓΠ, ΓΣ, ΘΤ
1.5.1	TK	Class- diagram- v0.1	1.4	10/5	14/5	28/5	6-7	
1.5.2	TK	Project Code	-	10/5	-	28/5	70-80	
1.5	M	D5	1.1, 1.2, 1.3, 1.4	-	28/5	28/5	76-87	ΑΛ, ΓΠ, ΓΣ, ΘΤ
1.6.1	TK	Team-plan- v1	1.5	29/5	30/5	4/6	1-2	

1.6.2	TK	Project-description-v1	1.5	29/5	30/5	4/6	1-2	
1.6.3	TK	Project-plan-v1	1.5	29/5	30/5	4/6	1-2	
1.6.4	TK	Risk-assessment-v1	1.5	29/5	30/5	30/5	1-2	
1.6.5	TK	Use-case-v1	1.6.4	31/5	1/6	1/6	2-3	
1.6.6	TK	Domain-model-v1	1.6.4	31/5	1/6	1/6	2-3	
1.6.7	TK	Robustness-diagrams-v1	1.6.6	2/6	3/6	4/6	2-3	
1.6.8	TK	Sequence-diagrams-v1	1.6.6	2/6	3/6	4/6	2-3	
1.6.9	TK	Class-diagram-v1	1.6.6	2/6	3/6	4/6	2-3	
1.6.10	TK	Project Code	-	29/5	-	4/6	30-40	
1.6	M	D6	1.1, 1.2, 1.3, 1.4, 1.5	-	4/6	4/6	44-62	ΑΛ, ΓΠ, ΓΣ, ΘΤ

Οι παρακάτω εικόνες δείχνουν τα γραφήματα Gantt και Pert που αντιστοιχούν στη πληροφορία που αναλύεται στον παραπάνω πίνακα.

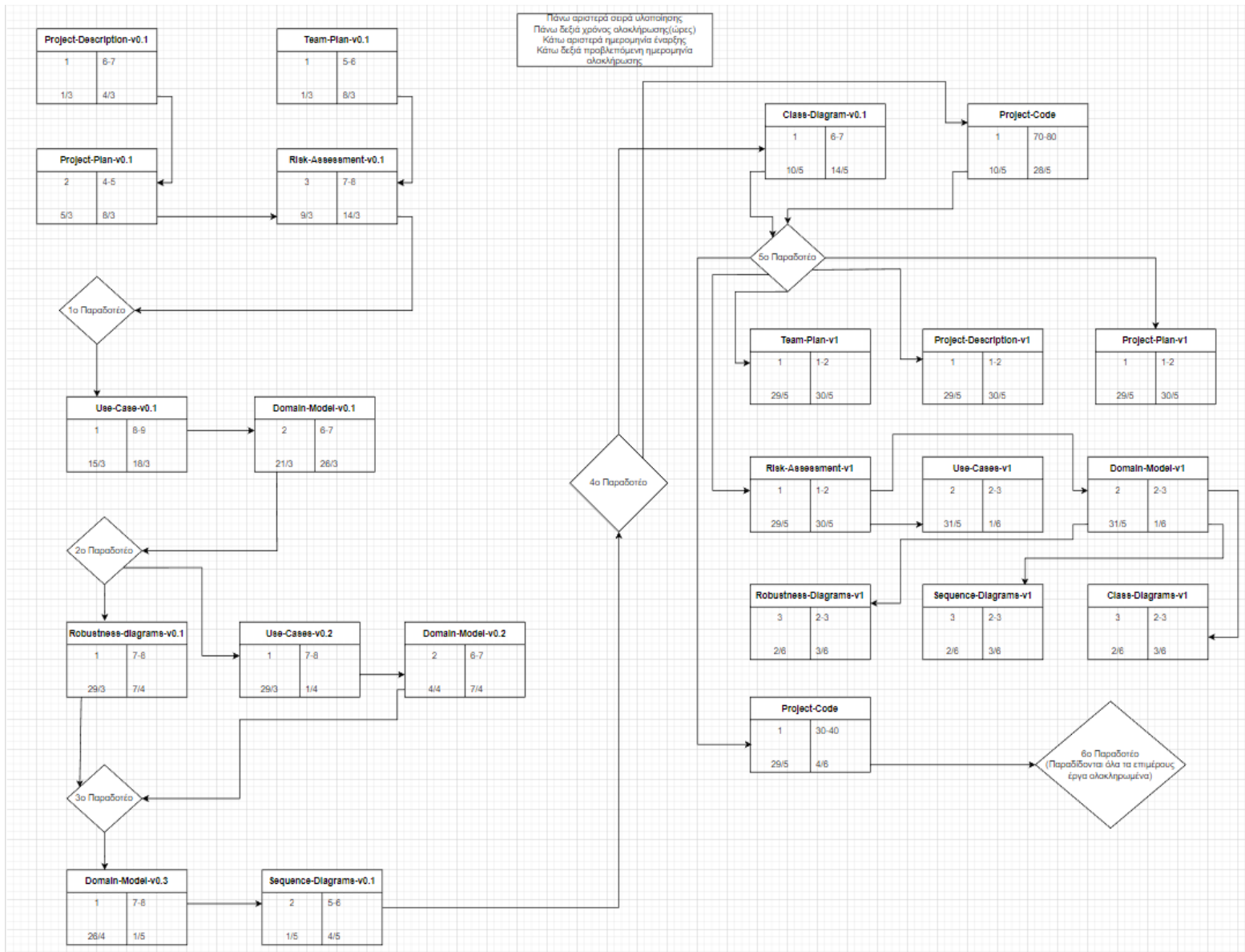
1.3.1. Gantt



1.3.2. Pert

Το ανάλογο Pert chart των παραδοτέων και εργασιών φαίνεται στην παρακάτω εικόνα.

Διευκρινήσεις για τα περιεχόμενα φαίνονται στο πλαίσιο στο πάνω μέρος της φωτογραφίας.



1.4. Εργαλεία

Τα εργαλεία που έχουν χρησιμοποιηθεί ή θα χρησιμοποιηθούν στο μέλλον χωρίζονται σε αυτά που θα χρησιμοποιηθούν για το software development κομμάτι και σε αυτά που θα χρησιμοποιηθούν για το management κομμάτι.

Εργαλείο		Χρήση
Git		VCS
Github		Repo hosting
Skype		Επικοινωνία ομάδας
Android Studio		Περιβάλλον για ανάπτυξη και testing της εφαρμογής
Kotlin		Ανάπτυξη της εφαρμογής
MS Word/LibreOffice		Συγγραφή τεχνικών κειμένων
GNU Planner		Χρονοπρογραμματισμός και δημιουργία Gantt chart.
Draw.io		Δημιουργία σχημάτων, γραφήματος Pert.
Figma		Mock-ups εφαρμογής

Πιο συγκεκριμένα:

1.4.1. Software Development

1. Git

Θα χρησιμοποιηθεί για version control στον κώδικα.

2. Github

Θα χρησιμοποιηθεί για τη δημιουργία ενός κοινού repository και την πρόσβαση σε αυτό από κάθε μέλος της ομάδας.

3. Skype

Η χρήση του θα γίνει κυρίως για την εξ-αποστάσεως οργάνωση του project και τα εβδομαδιαία meetings.

4. Android Studio

Το Android Studio θα χρησιμοποιηθεί ως βασικό IDE για την ανάπτυξη της εφαρμογής καθώς και το testing και packaging αυτής.

5. Kotlin

Η Kotlin αποτελεί ιδανική γλώσσα για ανάπτυξη εφαρμογών Android, και επίσης έχει καλή υποστήριξη από το Android Studio.

1.4.2. Management

1. MS Word/LibreOffice

Θα χρησιμοποιηθούν για τη συγγραφή τεχνικών κειμένων.

2. Figma

Θα χρησιμοποιηθεί για τη δημιουργία mock-ups της εφαρμογής πάνω σε οθόνες κινητού.

3. Draw.io

Θα χρησιμοποιηθεί για τη δημιουργία σχημάτων όπως των robustness diagrams κατά τη διάρκεια του project, και άλλων γραφημάτων όπως το Pert chart.

4. GNU Planner

Θα χρησιμοποιηθεί για τον χρονοπρογραμματισμό του κάθε παραδοτέου, τη δημιουργία του Gantt από αυτή τη πληροφορία, και την ευκολότερη οργάνωση των υπεύθυνων του κάθε παραδοτέου.