|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Covid-19 Gov Tracker | | | | |
|  |  | | |  |
| HLD-Services | | | | |
|  | | Mobile App |  | |

TABLE OF CONTENTS

[agenda 1](#_Toc101002779)

[I. Brief 1](#_Toc101002780)

[II. Description Scope 2](#_Toc101002781)

[III. Financial Scope 3](#_Toc101002782)

[IV. Focused on Delivery 3](#_Toc101002783)

[V. Project Execution 3](#_Toc101002784)

[VI. Technologic Approach 4](#_Toc101002785)

[1. DESIGN 6](#_Toc101002786)

[1.1. PROBLEM ANALYSIS 6](#_Toc101002787)

[1.2. PERSONA 6](#_Toc101002788)

[1.3. Requirement Analysis (non/functional) & Use Cases Ερμηνείες 7](#_Toc101002789)

[1.4. UML Representation Analysis 10](#_Toc101002790)

[1.4.1. Systems Use Case 10](#_Toc101002791)

[1.5. Class Diagrams & ERD 11](#_Toc101002792)

[1.6. Storyboards & Wireframes (low fidelity) 13](#_Toc101002793)

[1.7. User Flows 17](#_Toc101002794)

[1.8. State Machine/State Chart Diagram 18](#_Toc101002795)

[1.9. Site Map Design & Planning 18](#_Toc101002796)

[1.10. Heat Map/Mockup 19](#_Toc101002797)

[1.11. Testing 20](#_Toc101002798)

[2. Project Management 21](#_Toc101002799)

[2.1. Lifecycle 21](#_Toc101002800)

[2.2. Task/Time Schedule PERT 22](#_Toc101002801)

[2.3. Work Breakdown Structure (WBS) 23](#_Toc101002802)

[2.4. Sprint Canvas 24](#_Toc101002803)

[2.5. Scrum Cabinet Reports 24](#_Toc101002804)

[2.6. Project Participants – κατανομή εργασιών 25](#_Toc101002805)

[2.6.1. Stakeholder List 25](#_Toc101002806)

[2.6.2. Product Owner 25](#_Toc101002807)

[2.6.3. Scrum Master 26](#_Toc101002808)

[2.6.4. Scrum Team Members 27](#_Toc101002809)

[2.7. Project Initiation 27](#_Toc101002810)

[2.7.1. Project Charter 27](#_Toc101002811)

[2.7.1.1. Project Vision 27](#_Toc101002812)

[2.7.1.2. Project Mission 28](#_Toc101002813)

[2.7.1.3. Project Success Criteria 28](#_Toc101002814)

[2.7.2. Use Case Diagram & Report 28](#_Toc101002815)

[2.7.3. Prioritized Use Cases 28](#_Toc101002816)

[2.7.4. Epics 29](#_Toc101002817)

[2.8. Product Backlog – Features 29](#_Toc101002818)

[2.8.1. User Story Map 30](#_Toc101002819)

[2.8.2. Prioritized User Stories 30](#_Toc101002820)

[2.9. Project Deliverables – Release Planning 30](#_Toc101002821)

[2.10. Burnout Chart/Line 31](#_Toc101002822)

[Bibliography 32](#_Toc101002823)

[Figure 1 BPMN Business Model Plan Covid Tracker Process Oriented 2](#_Toc101003175)

[Figure 2 Product Backlog Topics Summary 4](#_Toc101003176)

[Figure 3 Project's Persona 7](#_Toc101003177)

[Figure 4 General Dev Mgmt. Overview 10](#_Toc101003178)

[Figure 5 Main Mobile System Use Case 11](#_Toc101003179)

[Figure 6 UML Class Diagram με cardinalities/πληθυκότητες 12](#_Toc101003180)

[Figure 7 Log in Screen – Wireframe 14](#_Toc101003181)

[Figure 8 Dashboard – Wireframe 15](#_Toc101003182)

[Figure 9 Add Patient UI – Wireframe 16](#_Toc101003183)

[Figure 10 Statistics Viz - Wireframe 17](#_Toc101003184)

[Figure 11 Basic Flowchart Android App 18](#_Toc101003185)

[Figure 12 State Machine Basic 18](#_Toc101003186)

[Figure 13 Site Map 19](#_Toc101003187)

[Figure 14 Main Dashboard Heat Areas (red is more eye tracked) 20](#_Toc101003188)

[Figure 15 Process Management Lifecycle 21](#_Toc101003189)

[Figure 16 ITSM Identification 22](#_Toc101003190)

[Figure 17 ITSM Initiation 22](#_Toc101003191)

[Figure 18 PERT timeline \*no critical path (CPM) for one-way-trip path 23](#_Toc101003192)

[Figure 19 Work Breakdown process overview 23](#_Toc101003193)

[Figure 20 Scrum Canvas 24](#_Toc101003194)

[Figure 21 Scrum Cabinet 24](#_Toc101003195)

[Figure 22 User Story Map 30](#_Toc101003196)

[Figure 23 Burndown chart Sprint 1 Y as Story points of Sprint No.1 == 30 task points 31](#_Toc101003197)

[Table 1 Stakeholder List 25](#_Toc101002881)

[Table 2 Product Owner 25](#_Toc101002882)

[Table 3 Scrum Master 26](#_Toc101002883)

[Table 4 Scrum Team Members 27](#_Toc101002884)

[Table 5 Prioritized Use Cases 28](#_Toc101002885)

[Table 6 Use Case to Epic Step 29](#_Toc101002886)

[Table 7 Prioritized User Stories 30](#_Toc101002887)

[Table 8 Project Deliverables 31](#_Toc101002888)

[Table 9 Release Configuration 31](#_Toc101002889)

# agenda

## Brief

* Project Name:
  + Covid Gov Tracker
* Project Description:
  + Ένας πελάτης επιθυμεί να αναπτύξει και να δοκιμάσει μια πλατφόρμα για να ενεργοποιήσει ένα διαδικτυακό ιογενές σύστημα παρακολούθησης για ασθενείς, να φιλοξενεί το δικό του περιεχόμενο στον δικό του τομέα.
* Who is this for?
  + Government – αυτό το προϊόν είναι για την κυβέρνηση που επιθυμεί να έχει έναν εύκολο τρόπο εντοπισμού Συστημάτων ή/και παρακολούθησης του ιστορικού τους και των επιπέδων ιού.
* Feature List (product requirements)
  + Easy onboarding
  + Viral Tracker, History/timeline
  + Patient Status, information
  + Personalization screen
  + Statistics Analysis και Visualization
* Deliverables
  + Wireframes για client approval
  + High fidelity prototype του product
  + User Testing
  + Usability Report
  + UI Assets για developers
* Timeline/Deadline
* Η διάρκεια περιγράφεται παρακάτω ως subtask

## Description Scope

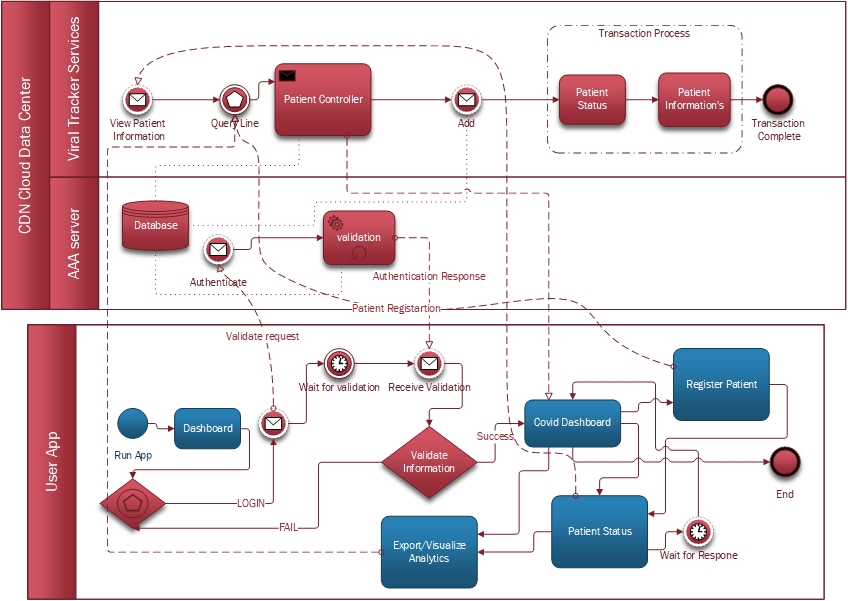


Figure BPMN Business Model Plan Covid Tracker Process Oriented

Ο σκοπός αυτού του εγγράφου είναι να παρέχει έγκυρες πληροφορίες για τα μέλη του, συμπεριλαμβανομένης της σχεδίασης υψηλού επιπέδου(*HLD*) σε μια συνοπτική εφαρμογή της εφαρμογής Περιεχόμενο Υγείας Ασθενών Covid Tracker. (Wikipedia, 2021)

Η εφαρμογή θα αποτελείται από λειτουργικότητα διεπαφής frontend & backend και διεπαφές εφαρμογών για κινητά. Θα επιτρέψει στον χρήστη να προσθέτει ασθενείς, να ανακτά λεπτομέρειες και να παρακολουθεί το ιστορικό, να παρακολουθεί την υγεία τους και να έχει μια ολοκληρωμένη σουίτα συμβάντων ελέγχου υγείας. Η εφαρμογή θα είναι ένα υποσύνολο ενός πληροφοριακού συστήματος που θα αλληλοεπιδρά με χρήστες και άλλα συστήματα βάσεων δεδομένων. (www.bpmn.org, n.d.)

## Financial Scope

Το έργο θα τηρεί τις προθεσμίες, τον προϋπολογισμό, το όραμα των ενδιαφερομένων.

* Επενδυτική απόφαση
* Απόφαση για το κεφάλαιο κίνησης
* Απόφαση χρηματοδότησης
* Απόφαση μερίσματος
* Εξασφαλίζει ρευστότητα
* Διαχείριση κερδών

## Focused on Delivery

*Subtasks/activities:*

1) Requirement’s analysis

2) Database/Business Entities design

3) Interface design

4) Database/ Business Entities development και data entry

5) Interface development

6) System completion

7) Application installation και control (integration)

8) Training και acceptance/support

## Project Execution

Διαθέσιμες προσεγγίσεις: procedural/waterfall, agile, dev-ops

Μια βάση εξειδίκευσης του προσωπικού και διαθέσιμη υποδομή είναι η επιλεγμένη προσέγγιση ευέλικτης ομάδας scrum. Η εγγραφή και διαχείριση του κύριου μαθήματος θα γίνει μέσο συνεργασίας και ευέλικτης πλατφόρμας JIRA με σύστημα παρακολούθησης και καταγραφής προβλημάτων/πηγή εισιτηρίων και έλεγχος έκδοσης DVCS GIT.

Product Backlog Overview

Παρακάτω ακολουθεί μια αρχική εκτίμηση η οποία θα αλλάξει δυναμικά (ενημέρωση / πέρασμα του χρόνου) με περισσότερες πληροφορίες καθώς θα αλλάξουν οι προτεραιότητες καθώς και οι απαιτήσεις του έργου ενδέχεται να αλλάξουν.



Figure 2 Product Backlog Topics Summary

\*Από την παραπάνω ενότητα Ημερήσια διάταξη συνοψίζονται τα πάντα, συμπεριλαμβανομένου του Gantt και backlog, αυτά δεν είναι καθόλου κοντά στις πραγματικές απαιτήσεις αλλά μια στενή γενικευμένη εκτίμηση του τι πρέπει να αναμένεται. Στη συνέχεια θα εισαχθούν πραγματικές υλοποιήσεις σε αυτό το έγγραφο.

## Technologic Approach

Η εφαρμογή θα είναι mobile based

1. Web tech & frameworks: Spring boot(back-end) + Java Native Android για Frontend.
2. Spring Security
3. ORM: JPA Hibernate
4. Database System: MySQL
5. Third-Party libraries:
   * + - 1. spring-boot-starter-web
         2. spring-boot-starter-test
         3. spring-boot-starter-jdbc
         4. spring-boot-starter-data-jpa
         5. mysql-connector-java
         6. spring-boot-starter-security
         7. commons-lang3

# DESIGN

## PROBLEM ANALYSIS

Πρόβλημα:

«Να γίνει ο σχεδιασμός για μια εφαρμογή Android για κινητό, η οποία θα εξυπηρετεί την καταγραφή κρουσμάτων COVID-19. Την εφαρμογή θα την χειρίζεται ένας υπάλληλος του υπουργείου προστασίας του πολίτη. Σε αυτή θα καταγράφονται τόσο τα στοιχεία του κάθε χρήστη, όσο και η τοποθεσία διαμονής 3 του, αλλά και το χρονικό συμβάν. Επιπλέον, η εφαρμογή θα είναι σε θέση να παράγει στατιστικά για τα δεδομένα που διαθέτει.»

Σύνοψη Λύσης:

Μια εφαρμογή καταγραφής κρουσμάτων για κινητό συγκεκριμένα Android Platform θα χρειαστεί back-end service & Controllers με μία βάση δεδομένων για Non-volatile αποθήκευση δεδομένων που σημαίνει ότι η εφαρμογή θα λειτουργεί σαν front-end θα την χειρίζεται ο εκάστοτε χρήστης ειδικής εκπαίδευσης από τον οργανισμό/κυβέρνηση καθώς αυτός θα εισάγει δεδομένα για νέους χρήστες/ασθενείς διάφορες πληροφορίες αυτές θα στέλνουν αίτημα στο back-end service ώστε να εκχωρηθεί άδειας αποθήκευσης τους σε μόνιμα μέρη αποθηκεύσεις δομής. Ώστε αυτά τα δεδομένα να μπορούν να αναλυθούν ποιοτικά και ποσοτικά από ερευνητικού επιστήμονες αλλά και γενική στατιστική ανάλυση για την κλίμακα επιπέδων ιογενείς υπάρξεις στην χώρα.

Η φυσική υποδομή θα πρέπει να λάβει χώρα σε Data Centers / (private) Cloud καθώς θα ώστε έτσι να τηρηθεί ή μη-αμελητέα αποθήκευση δεδομένων για να υπάρξει Confidentiality, availability, reliability, Integrity

Η χρήση τεχνολογιών αναφέρεται εδώ -> Technologic Approach

## PERSONA

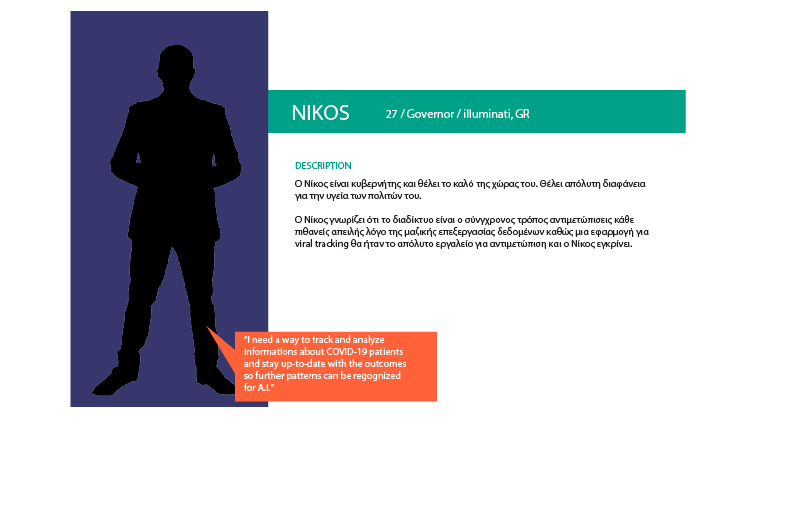


Figure 3 Project's Persona

Το project θα επεκταθεί όπως θέλει ο Νίκος (Faller, 2019).

## Requirement Analysis (non/functional) & Use Cases Ερμηνείες

indicative specification functional requirements

**Function Name – Use Case:** Account Login

**Short Description:** Ο Υπάλληλος του υπουργείου προστασίας του πολίτη θα συνδέεται στο σύστημα.

**Conditions:** Μόνο έγκυρη λογαριασμοί από το κατάλληλο domain/department.

**Inputs:** Front-end portal authenticated στο back-end

**System behavior – Main Flow:** Το σύστημα λαμβάνει Δεδομένα από το mobile app συγκρίνοντας κάθε μοναδικό χρήστη ξεχωριστά με βάση τα πεδία εισαγωγής του. Ο «χρήστης» ζητά ένα διακριτικό στην ασφάλεια Spring, εάν παρείχε σωστό ένα JWT που επιστρέφει για να αποκτήσει πρόσβαση στη λειτουργικότητα RESTful API to-do.

**Alternative Flow:** -

**Output – Exit Status:** Πρόσβαση σε χρήση Services του συστήματος.

**Function Name – Use Case:** Accounts & ΑΑΑ (*authentication, authorization, accounting*)

**Short Description:** Πολλαπλοί είδη λογαριασμού Role-based access control (*RBAC*).

**Conditions:** Έλεγχος και έναρξη καταγραφής επιπέδου πρόσβασης και δραστηριότητας κατά την σύνδεση.

**Inputs:** Front-end portal authenticated στο back-end.

**System behavior – Main Flow:** Κατά την είσοδο στο σύστημα ελέγχεται το επίπεδο πρόσβασης στο back-end με βάση τα εισαγόμενα στοιχεία χρήστη που αναθέτη ένα RBAC αριθμό καθώς και services για AAA enabled.

**Alternative Flow:** -

**Output – Exit Status:** Διαφοροποίηση & Έλεγχος πρόσβασης σε χρήση Services του συστήματος.

**Function Name – Use Case:** Patient Registry

**Short Description:** Ο συνδεμένος χρήστης διαχειρίζεται νέους/παλιούς ασθενής.

**Conditions:** Καταχώρηση ασθενών εφόσον βγουν θετικοί μέσο τεστ.

**Inputs:** Mobile user interface

**System behavior – Main Flow:** Ο χρήστης εισάγει με βάση το RBAC του πληροφορίες στην βάση δεδομένων με βάση τους ασθενής και περίοδο νοσήσεις.

**Alternative Flow:** -

**Output – Exit Status:** Καταχώρηση εγγραφών χρήστη και ενημέρωση των στοιχείων/εγγραφών τους.

**Function Name – Use Case:** Export Statistical analysis.

**Short Description:** Ο διαχειριστής εισάγει ή αφαιρεί μια καταχώρηση από το σύστημα.

**Conditions:** Database <> {} and patient <> [].

**Inputs:** Mobile user interface.

**System behavior – Main Flow:** Το σύστημα παράγει αντλώντας δεδομένα τις βάσεις δεδομένων σε δομή δεδομένων και τα σκιαγραφεί εικονικά μετέπειτα από αλγοριθμική ανάλυση τους στην εφαρμογή κινητού για την πορεία, πλήθος, κ.α., περί νοσήσεις.

**Alternative Flow:** -

**Output – Exit Status:** Statistical visualization output για Data analysis.

**Function Name – Use Case:** Admin CRUD’s Database.

**Short Description:** Ο διαχειριστής εισάγει ή αφαιρεί μια καταχώρηση από το σύστημα.

**Conditions:** RBAC login Admin Level.

**Inputs:** Front-end page admin account.

**System behavior – Main Flow:** Το σύστημα είναι σε θέση να αφαιρέσει ή να προσθέσει μια νέα καταχώρηση με βάση τις ενέργειες λειτουργίας διαχειριστή.

**Alternative Flow:** -

**Output – Exit Status:** Business entity database update.

indicative specification non-functional requirements

**Function Name:** Portal differences on login

**Function Name:** Fast και responsive Database logging system

**Function Name:** Realtime updates

**Function Name:** ssl(https), pki connections

**Function Name:** Load balancing network server offload

**Function Name:** Data center private cloud setup

**Function Name:** Esxi Vmotion enabled

**Function Name:** CIA (Confidentiality, integrity and availability) triad

## UML Representation Analysis

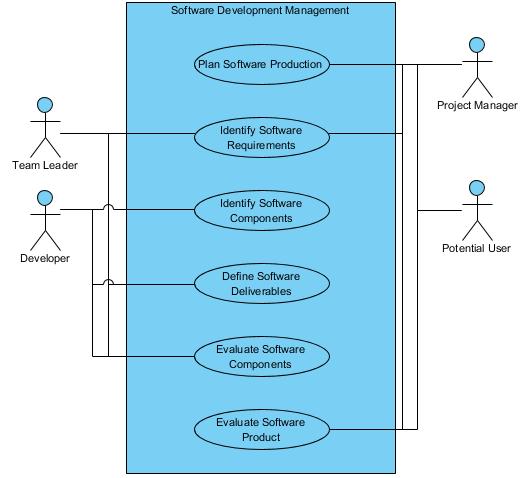


Figure General Dev Mgmt. Overview

## Systems Use Case

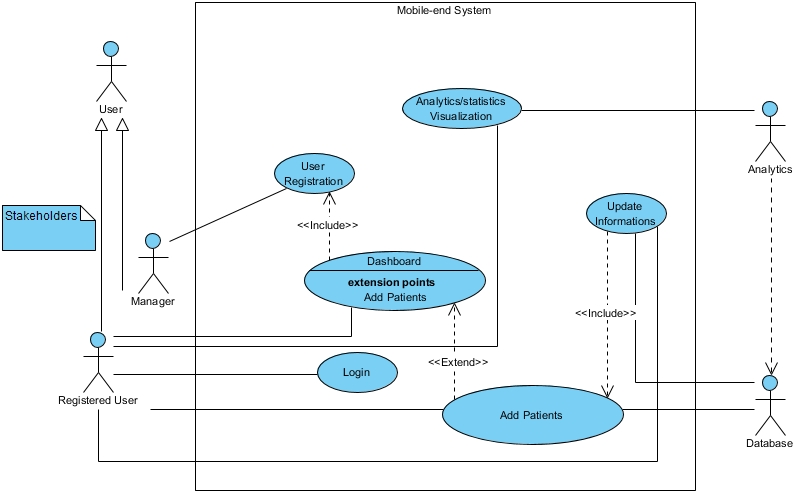


Figure Main Mobile System Use Case

## Class Diagrams & ERD

Λόγο χρήσης ORM (*object relational mapping*) τεχνολογιών το παρακάτω διάγραμμα μπορεί να χρησιμοποιηθεί και ως ERD. (hibernate.org, n.d.; Visual-paradigm.com, 2011; Paradigm, 2019)

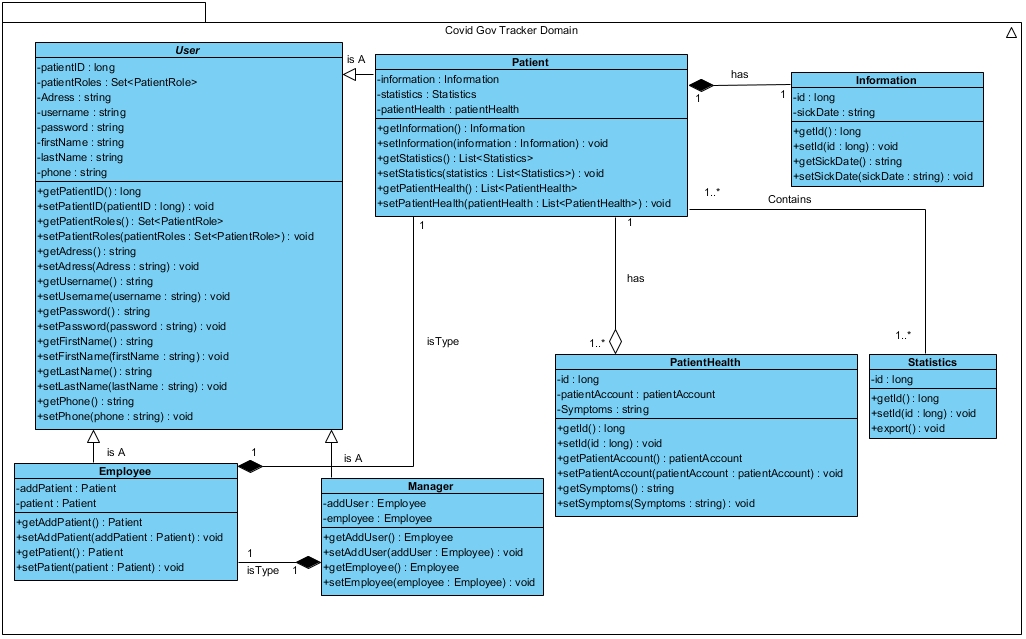


Figure UML Class Diagram με cardinalities/πληθυκότητες

Στην παραπάνω λίστα κλάσεων όλοι είναι και patients έστω και με empty history, η class User είναι abstract που σημαίνει δεν δημιουργούμε objects από αυτήν μόνο κληρονομούμε (Oracle, 2019).

Ο Admin δεν υπάρχει σαν κλάση διότι είναι RBAC 0 (*Role Based Access Control*) και κάνει Login μέσω web browser να προσθέσει Managers. (docs.oracle.com, n.d.)

Οι Manager είναι ταυτόχρονα και employee όταν δημιουργείται δηλαδή ένας Manager ⬄ Employee έτσι έχει πρόσβαση σε όλες τις μεθόδους ενός employee, επίσης αφού είναι και patient αυτό του δίνει πρόσβαση στην statistics που είναι μια List/Sequence από όλους τους ασθενής (*εφόσον την εφαρμογή την χειρίζεται μόνο εξειδικευμένο προσωπικό και δεν είναι για όλους άρα μπορούμε με ασφάλεια να δηλώσουμε μεθόδους και στην κλάση Patient που θα ήταν πιο high Privileged*).

Οι employees προσθέτουν και κάνουν Update άλλους Patients εφόσον έχουν πλήρη πρόσβαση λόγο composition στην class patient θέτοντας CRUD requests προς τον back-end service server controller.

Ένας patient έχει πολλά patient health διότι είναι incremental/record κάθε περίπτωση νοσήσεις διότι μπορεί να νοσήσει πολλαπλές φορές άρα θέλουμε δομή record και όχι όλα σε 1 προς 1 αλλά 1 προς πολλά από άποψη patient => patientHealth, επίσης ο patient μπορεί να υπάρχει αλλά δεν σημαίνει ότι έχει νοσήσει λόγο την ιεραρχία του συστήματος ότι ένας manager & User είναι και patient έστω και με empty history άρα χρειαζόμαστε να μην είναι δεσμευμένο το History ⬄ patientHealth που σηματοδότη την νοσήσει.

Η κλάση statistics έχει μεθόδους και properties-objects τύπου List<Patient> get & set (*λόγο many-to-many*) ωστόσο στο σχεδιάγραμμα δεν φαίνεται διότι θελήσαμε να δηλώσουμε την αποφύγει json loop έτσι παραλήφθηκε η δήλωση τους. Επίσης λόγο “double” composition μπήκε το πιο general class relationship “association”.

## Storyboards & Wireframes (low fidelity)

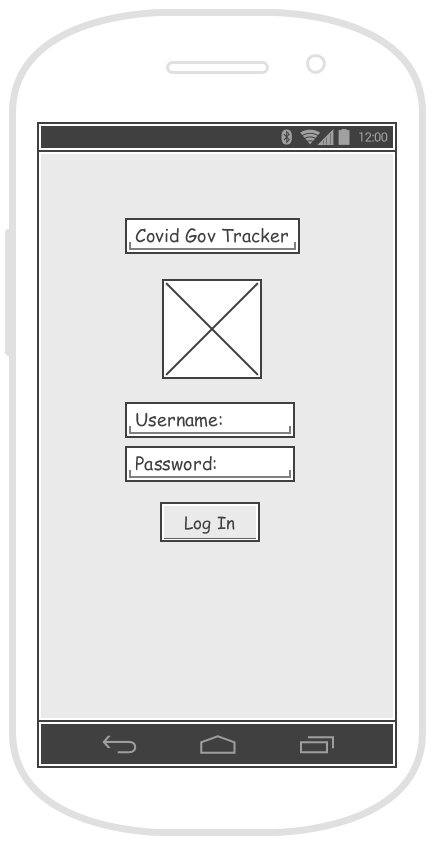


Figure Log in Screen – Wireframe

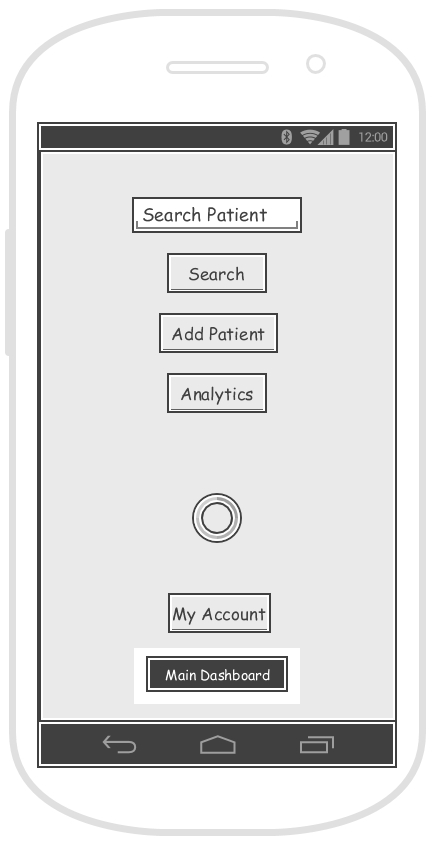


Figure Dashboard – Wireframe



Figure Add Patient UI – Wireframe

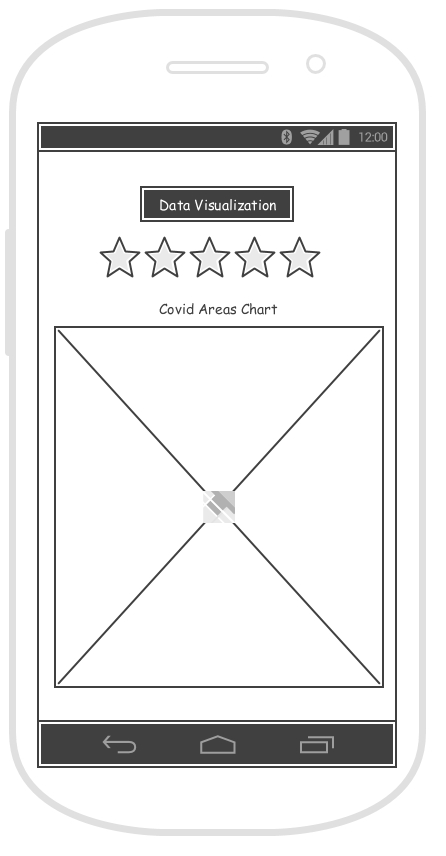


Figure Statistics Viz - Wireframe

## User Flows

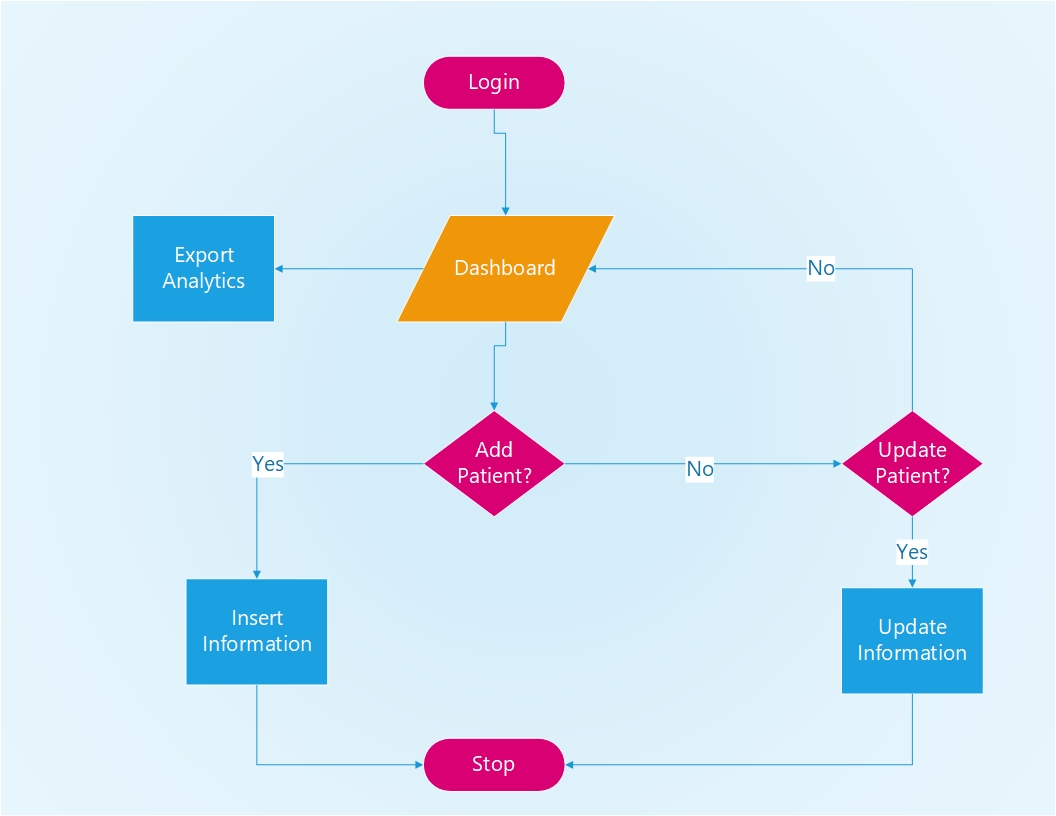


Figure Basic Flowchart Android App

## State Machine/State Chart Diagram

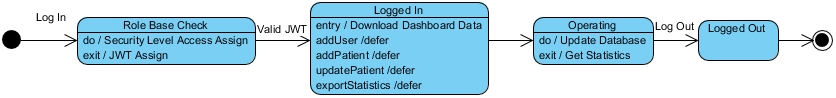


Figure State Machine Basic

## Site Map Design & Planning

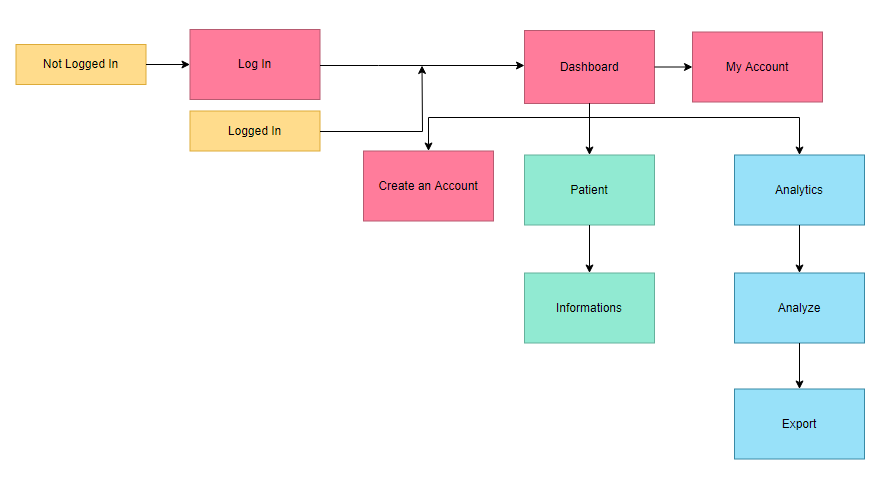


Figure 13 Site Map

Κάθε rectangular section είναι και ένα object και υπάρχουν διαθέσιμες η όχι κάποιες λειτουργείες ανάλογα με το RBAC.

## Heat Map/Mockup

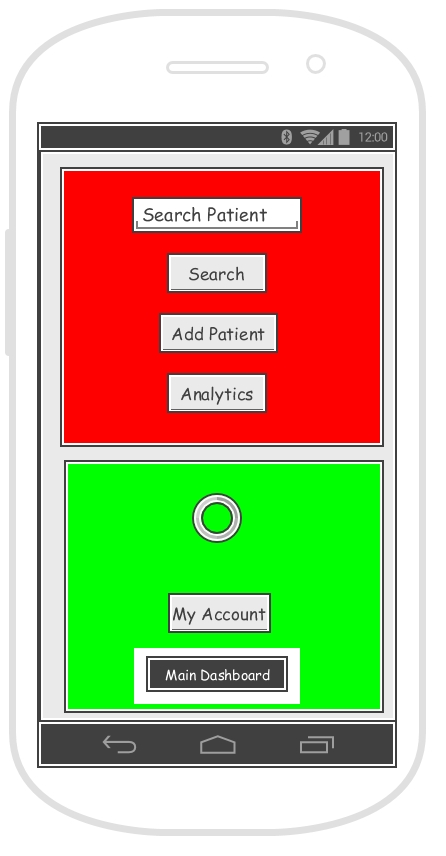


Figure Main Dashboard Heat Areas (red is more eye tracked)

Θα πρέπει να αναπτυχθούν διάφορα πρότυπα δοκιμών και σχεδιασμού για τη μέτρηση της προσβασιμότητας και της προτίμησης των χρηστών.

Ποσοστό εγκατάλειψης (bounce rate), Φύλο, Ηλικία, Συσκευή, Ενδιαφέροντα, Δοκιμές A/B + πραγματικοί χρήστες μέσω του Analytics. (Firebase, n.d.)

## Testing

Για την σωστή διεκπεραίωση των λειτουργειών εφαρμογής θα χρησιμοποιηθεί η τακτική Test Driven Development (*TDD*) στην οποία πρώτα γράφεις των κώδικα του testing και τι κάνεις σαν expect και ύστερα γράφεις των κώδικα του implementation σε συνδυασμό με ένα CI/CD e.g., Jenkins θα μπορεί να πραγματοποιηθεί σε κάθε commit GitHub real-time checking άμα έγινε αποδεκτή η εφαρμογή στα testing πριν/μετά για σωστό deploy σε containers Kubernetes, docker.

Στην συγκεκριμένη τεχνολογία Spring Boot το testing γίνεται μόνο μέσω χρήσης Dependency Injection ώστε να επιτραπεί το Unit testing καθώς και loggers τόσο στο front-end (native android) όσο και στο back-end (spring).

# Project Management

## Lifecycle

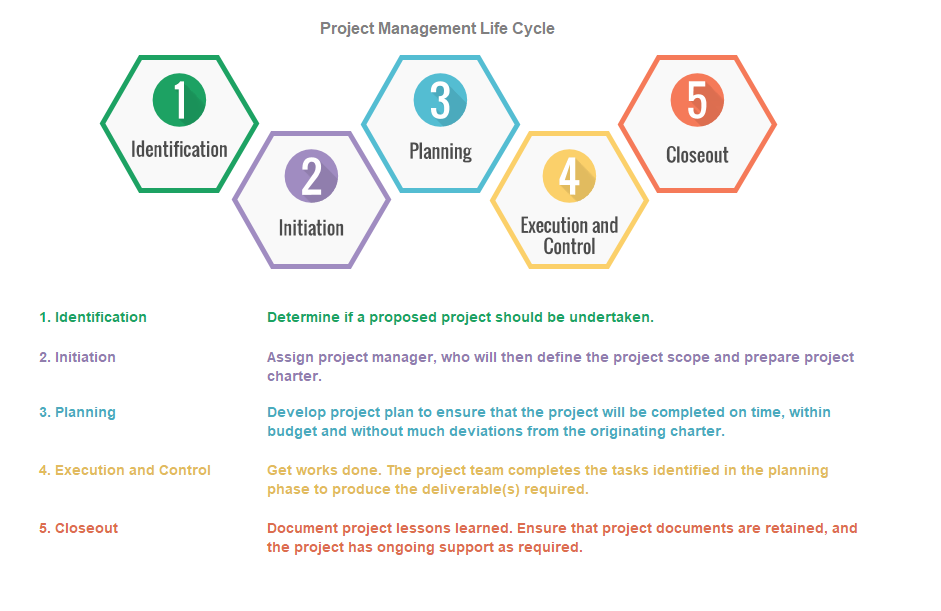


Figure Process Management Lifecycle

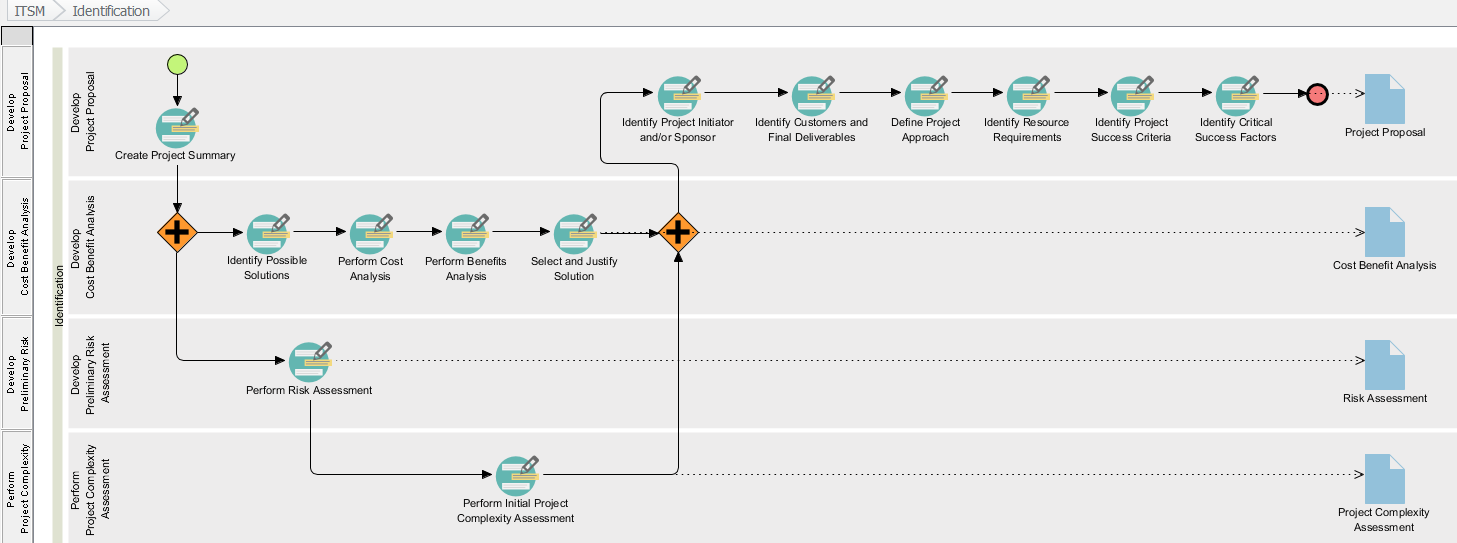


Figure ITSM Identification

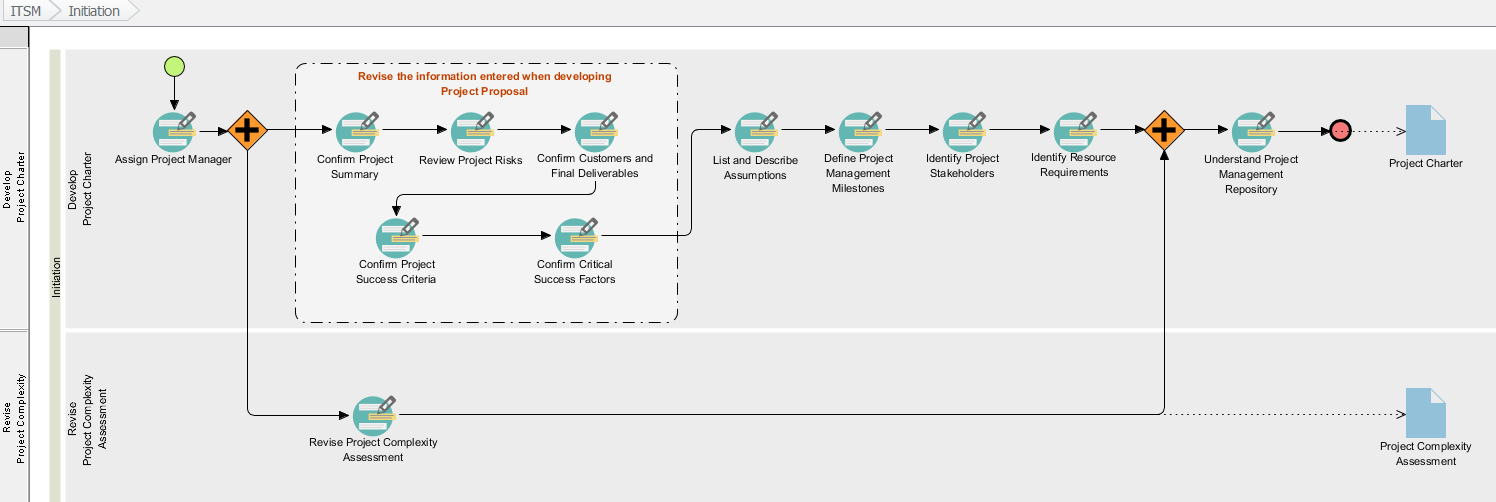


Figure ITSM Initiation

## Task/Time Schedule PERT

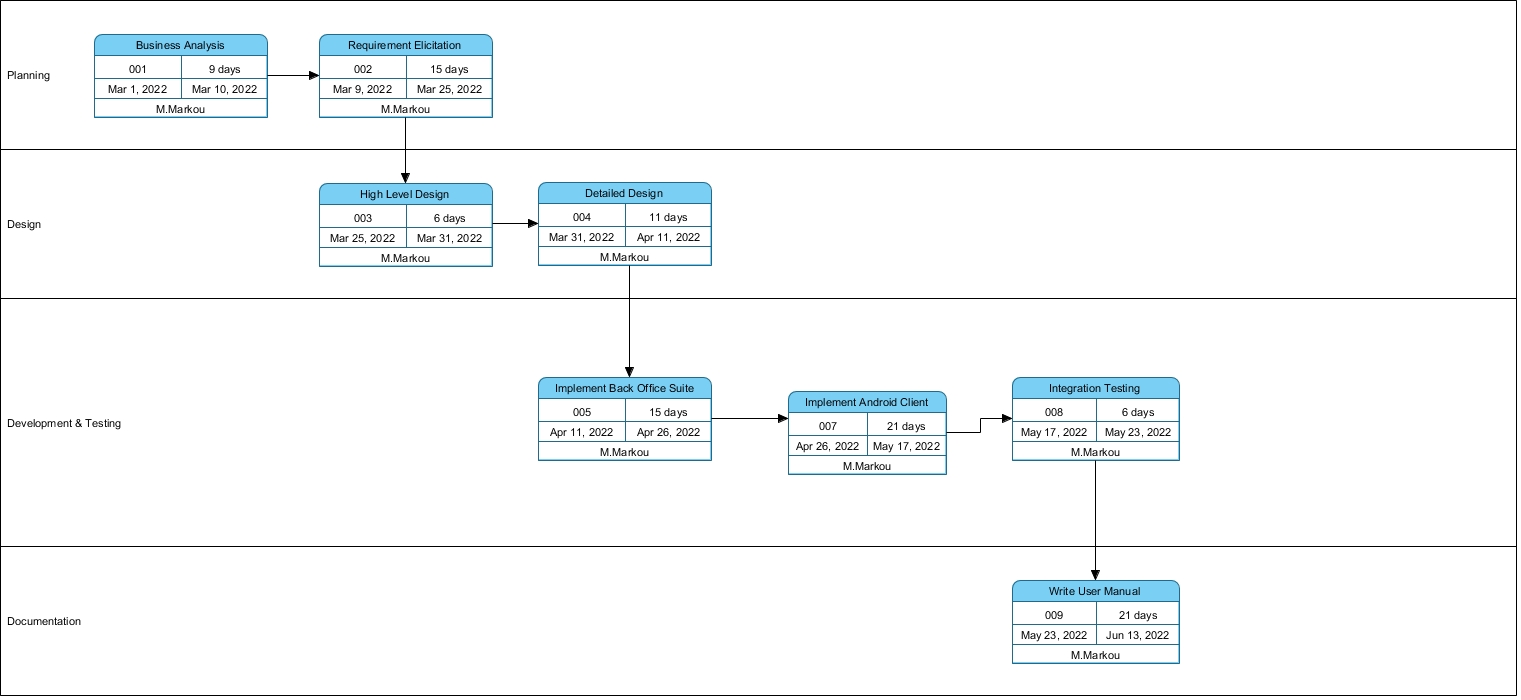


Figure PERT timeline \*no critical path (CPM) for one-way-trip path

## Work Breakdown Structure (WBS)

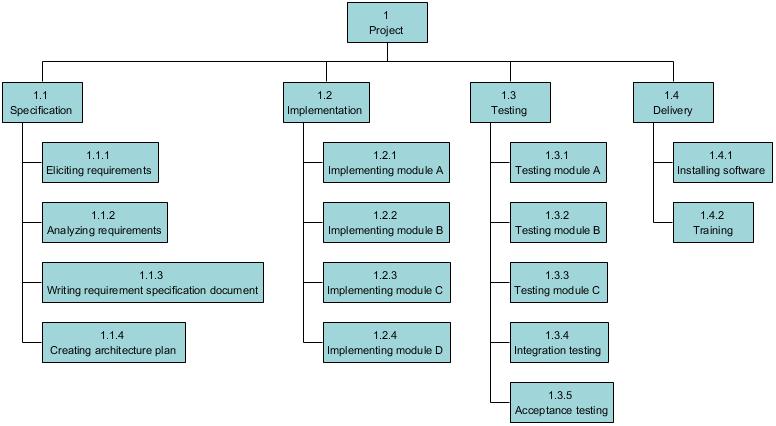


Figure Work Breakdown process overview

## Sprint Canvas

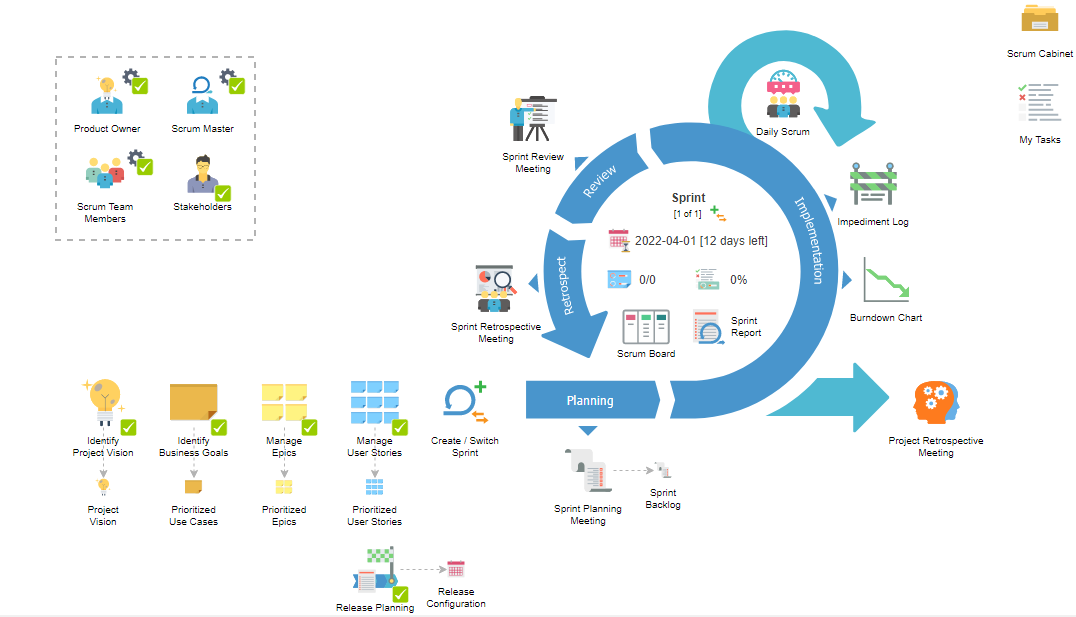


Figure Scrum Canvas

(Atlassian, 2019; Scrum.org, 2019)

## Scrum Cabinet Reports

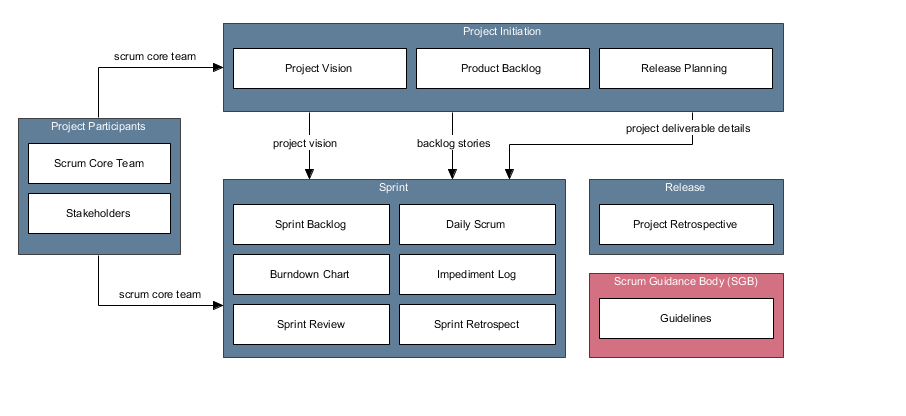


Figure Scrum Cabinet

## Project Participants – κατανομή εργασιών

* + 1. Stakeholder List

| Name | Role | Responsibilities |
| --- | --- | --- |
| Michail Markou | Sponsor, Customer, User | Διαχειρίζεται το τελικό αποτέλεσμα και το όραμα του έργου |

Table Stakeholder List

* + 1. Product Owner

| Member | Michail Markou |
| --- | --- |
| Responsibilities | * Καθορίζει το όραμα του έργου * Βοηθά στη δημιουργία του Χάρτη του Έργου * Προσδιορίζει τα ενδιαφερόμενα μέρη του έργου * Βοηθά στον προσδιορισμό των μελών της ομάδας scrum * Δημιουργήστε Epics * Δημιουργήστε, ορίστε και δώστε προτεραιότητα σε ιστορίες χρηστών * Δημιουργήστε σχέδιο έκδοσης και κρατήστε το ενημερωμένο * Έγκριση ιστοριών χρηστών * Εξηγεί τις ιστορίες των χρηστών και διευκρινίζει τις απαιτήσεις για την ομάδα scrum στη συνάντηση προγραμματισμού sprint * Παρέχει καθοδήγηση και διευκρίνιση στην εκτίμηση της προσπάθειας για εργασίες * Grooms prioritized backlog * Αποδοχή/απόρριψη παραδοτέων/deliverables * Παρέχετε σχόλια στις ομάδες scrum master και scrum team * Βοηθήστε την ανάπτυξη εκδόσεων προϊόντων και συντονίστε το με τον πελάτη * Συμμετοχή σε έντυπη αναδρομική συνάντηση |

Table Product Owner

* + 1. Scrum Master

| Member | Michail Markou |
| --- | --- |
| Responsibilities | * Βοηθά στον εντοπισμό των ενδιαφερομένων στο έργο * Διευκολύνει τη συγκρότηση της ομάδας scrum * Διευκολύνει τη δημιουργία επών * Βοηθά τον ιδιοκτήτη του προϊόντος να δημιουργήσει και να διατηρήσει το product backlog * Συντονίζει τη δημιουργία του Σχεδίου Έκδοσης * Βοηθά στη δημιουργία και τον καθορισμό ιστοριών χρηστών * Διευκολύνει τις συναντήσεις * Διευκολύνει την ομάδα scrum στη δημιουργία εργασιών για το επόμενο sprint * Διευκολύνει την ομάδα scrum στην εκτίμηση προσπάθειας * Υποστηρίζει την ομάδα scrum στη δημιουργία παραδοτέων * Βοηθά στη διατήρηση του αρχείου καταγραφής εμποδίων/impediment log * Διασφαλίζει ότι ανακαλύπτονται και επιλύονται ζητήματα που επηρεάζουν την ανάπτυξη |

Table Scrum Master

* + 1. Scrum Team Members

| Member | Responsibilities |
| --- | --- |
| Michail Markou | * Εξασφαλίζει σαφή κατανόηση των επών και των ιστοριών των χρηστών * Συμφωνεί με τα άλλα μέλη της ομάδας για το μήκος του sprint * Ζητά διευκρινίσεις σχετικά με νέες ενημερώσεις στις απαιτήσεις * Παρέχει στοιχεία στον ιδιοκτήτη του προϊόντος για τον καθορισμό και την εκτίμηση των ιστοριών χρηστών * Δεσμεύει τις ιστορίες των χρηστών να γίνουν σε ένα sprint * Παρέχει στοιχεία για τη δημιουργία και την εκτίμηση εργασιών * Αναπτύσσει το προϊόν, την υπηρεσία ή άλλα αποτελέσματα * Εντοπίζει τον κίνδυνο και εφαρμόζει ενέργειες μετριασμού του κινδύνου, εάν απαιτείται * Παρέχει εισόδους για την ενημέρωση του αρχείου καταγραφής εμποδίων/impediment log * Συζητά την πρόοδο και τα θέματα στην καθημερινή συνάντηση scrum * Επιδεικνύει ολοκληρωμένα παραδοτέα στην αναθεώρηση sprint * Προτείνει ευκαιρίες βελτίωσης σε αναδρομική συνάντηση sprint * Συμμετέχει στην αναδρομική συνάντηση του έργου |

Table Scrum Team Members

## Project Initiation

* + 1. Project Charter
       1. Project Vision

Σύστημα για χρήση ανίχνευσης καταγραφής κρουσμάτων καθώς και στατιστικής ανάλυσης των δεδομένων για έρευνες αλλά και πορεία κρατικής υγείας. Με ευκολία χρήσεις από mobile devices.

* + - 1. Project Mission

Διευκόλυνση καταγραφής κρουσμάτων και σύγκριση παλιών με νέων καθώς και στατιστική πορεία της χώρας. Ταυτόχρονη διευκόλυνση με on-demand Patient Registry μέσο κινητού. Stay up-to-date.

* + - 1. Project Success Criteria
* Έγκυρη καταγραφή κρουσμάτων και real-time update κατάστασης
* Στατιστική ανάλυση πορείας δεδομένων και πρόληψης ιογενών περιοχών
* Μείωση των κρουσμάτων με καθοδήγηση ασθενών και μη.
  + 1. Use Case Diagram & Report

(βλ. Figure 5)

* + 1. Prioritized Use Cases

| Name | Description | Priority | Size | Complexity |
| --- | --- | --- | --- | --- |
| Dashboard | Κεντρικό μενού ελέγχου ενεργειών | Must | Medium | Medium |
| User Registration | Προσθήκη χρήστη στο σύστημα | Must | Large | High |
| Login | Σύνδεση χρήστη στο σύστημα | Must | Large | Medium |
| Analytics/statistics Visualization | Γραφική αναπαράσταση πληροφοριών και απόκτηση γνώσεις | Should | Very Large | High |
| Update Information’s | Ενημέρωση κατάστασης ασθενών | Must | Medium | Low |

Table Prioritized Use Cases

* + 1. Epics

| Name | Parent Use Case | Priority | Risk |
| --- | --- | --- | --- |
| Registered User add Patients in the Database | Add Patients | Must | Low |
| Export analytics to User Interface | Analytics/statistics Visualization | Should | High |
| User RBAC System Access to resources | Login | Must | Low |
| User Updates Information about Patients | Update Information’s | Must | Medium |
| User Main Navigation to system resources | Dashboard | Must | Medium |

Table Use Case to Epic Step

## Product Backlog – Features

* + 1. User Story Map

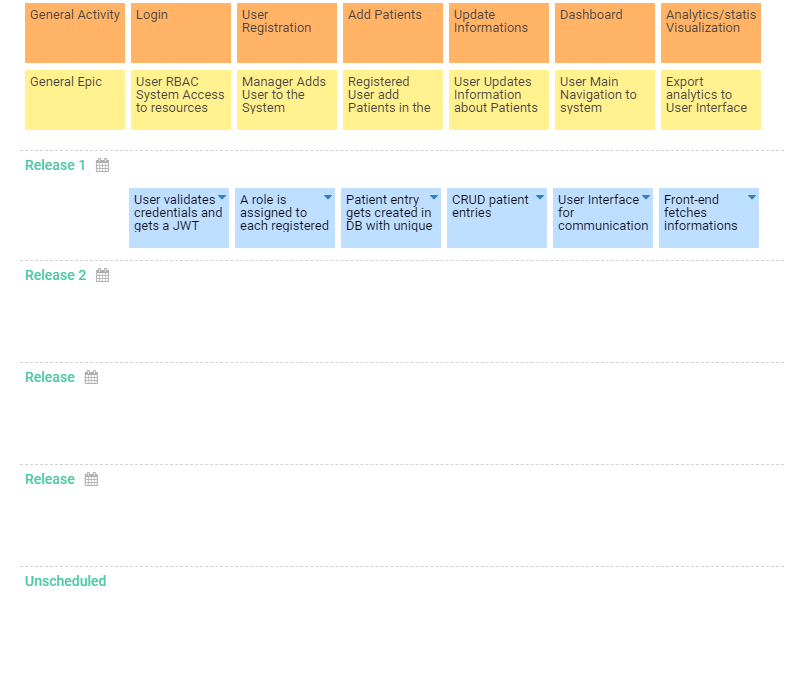


Figure User Story Map

* + 1. Prioritized User Stories

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Name** | **Epic** | **Status** | **Story Points** | **Priority** |
| A role is assigned to each registered user | Manager Adds User to the System | Approved | 3 | Must |
| User validates credentials and gets a JWT from backend | User RBAC System Access to resources | Approved | 5 | Must |
| Patient entry gets created in DB with unique ID | Registered User add Patients in the Database | Approved | 3 | Must |
| CRUD patient entries | User Updates Information about Patients | Approved | 1 | Must |
| User Interface for communication to back-end | User Main Navigation to system resources | Approved | 3 | Must |
| Front-end fetches information’s through back-end from the database to display knowledge | Export analytics to User Interface | New | 8 | Should |

Table Prioritized User Stories

## Project Deliverables – Release Planning

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Deliverable | Description | Priority | Status | Owner |
| Covid Gov Tracker (Android) | The Android Version of Front-end communication to the system | High | In Progress | Michail Markou |

Table Project Deliverables

|  |  |  |  |
| --- | --- | --- | --- |
| Release | Description | Release | Description |
| Release 1 | Support Major Features for Patient Registry and Tracking | Release 1 | Support Major Features for Patient Registry and Tracking |

Table Release Configuration

## Burnout Chart/Line

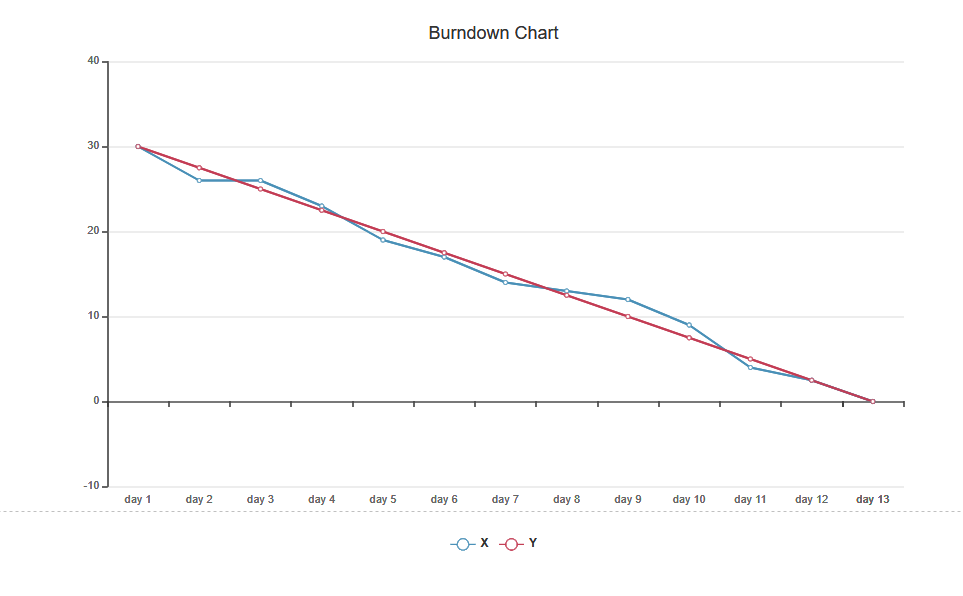


Figure Burndown chart Sprint 1 Y as Story points of Sprint No.1 == 30 task points

# Bibliography

Atlassian, 2019. *Agile best practices and tutorials | Atlassian.* [Online]   
Available at: https://www.atlassian.com/agile  
[Accessed 01 April 2022].

docs.oracle.com, n.d. *Configure Role-Based Access Control (RBAC).* [Online]   
Available at: https://docs.oracle.com/cd/E65459\_01/admin.1112/e65449/content/general\_rbac.html  
[Accessed 01 April 2022].

Faller, P., 2019. *What Are User Personas and Why Are They Important? | Adobe XD Ideas.* [Online]   
Available at: https://xd.adobe.com/ideas/process/user-research/putting-personas-to-work-in-ux-design/  
[Accessed 01 April 2022].

Firebase, n.d. *Firebase A/B Testing | Optimize your app experience through experimentation.* [Online]   
Available at: https://firebase.google.com/products/ab-testing  
[Accessed 01 April 2022].

hibernate.org, n.d. *Your relational data. Objectively. - Hibernate ORM..* [Online]   
Available at: https://hibernate.org/orm/  
[Accessed 01 april 2022].

Oracle, 2019. *Abstract Methods and Classes (The JavaTM Tutorials > Learning the Java Language > Interfaces and Inheritance).* [Online]   
Available at: https://docs.oracle.com/javase/tutorial/java/IandI/abstract.html  
[Accessed 01 April 2022].

Paradigm, V., 2019. *UML Class Diagram Tutorial.* [Online]   
Available at: https://www.visual-paradigm.com/guide/uml-unified-modeling-language/uml-class-diagram-tutorial/  
[Accessed 01 April 2022].

Scrum.org, 2019. *What is Scrum?.* [Online]   
Available at: https://www.scrum.org/resources/what-is-scrum  
[Accessed 01 April 2022].

Visual-paradigm.com, 2011. *How to Design Relational Database with ERD?.* [Online]   
Available at: https://www.visual-paradigm.com/tutorials/databasedesign.jsp  
[Accessed 01 April 2022].

Wikipedia, 2021. *High-level design.* [Online]   
Available at: https://en.wikipedia.org/wiki/High-level\_design  
[Accessed 01 April 2022].

www.bpmn.org, n.d. *BPMN Specification - Business Process Model and Notation.* [Online]   
Available at: https://www.bpmn.org/  
[Accessed 01 April 2022].