Έγγραφο απαιτήσεων λογισμικού (SRS)

ΠΡΟΣΑΡΜΟΓΗ ΤΟΥ ΑΝΤΙΣΤΟΙΧΟΥ ΕΓΓΡΑΦΟΥ ΤΟΥ ΠΡΟΤΥΠΟΥ ISO/IEC/IEEE 29148:2011

TollsIO

# Εισαγωγή

## 1.1 Εισαγωγή: σκοπός του λογισμικού

Το παρόν έγγραφο αποτελεί την Προδιαγραφή Απαιτήσεων Λογισμικού (SRS) για την ανάπτυξη ενός πληροφοριακού συστήματος στο οποίο θα έχουν πρόσβαση τα εμπλεκόμενα μέρη (stakeholders), και θα εξασφαλίζει την διαλειτουργικότητα στα διόδια της χώρας. Συγκεκριμένα, το λογισμικό θα δέχεται τα δεδομένα εισόδου, δηλαδή τις διελεύσεις, από τους λειτουργούς των αυτοκινητοδρόμων και θα τα αποθηκεύει προκειμένου να υπολογίζει ανά διαστήματα τις οφειλές κάθε λειτουργού σε κάθε άλλον, να τις συμψηφίζει και να δίνει την δυνατότητα στους λειτουργούς για την εξόφλησή τους, να αναλύει αυτά τα δεδομένα αλλά και να τα διαθέτει σε τρίτα μέρη για περαιτέρω επεξεργασία.

## 1.2 Διεπαφές (interfaces)

### 1.2.1 Διεπαφές με εξωτερικά συστήματα

Η εφαρμογή διασυνδέεται με εξωτερικά πληροφοριακά συστήματα για την ανταλλαγή και επεξεργασία δεδομένων. Οι κύριες διασυνδέσεις περιλαμβάνουν:

* Λειτουργούς των διοδίων
* Παρόχους υπηρεσιών πληρωμών

Η ανταλλαγή δεδομένων γίνεται μέσω RSTful APIs, που υποστηρίζουν

* Πρότυπα ανταλλαγής δεδομένων: JSON για δομημένη μεταφορά δεδομένων
* Πρωτόκολλα ασφαλείας: Όλα τα δεδομένα μεταφέρονται μέσω κρυπτογραφημένων HTTPS, εξασφαλίζοντας την ακεραιότητα και το απόρρητο των πληροφοριών.

### 1.2.2 Διεπαφές με το χρήστη

Η εφαρμογή παρέχει δύο βασικές διεπαφές για την αλληλεπίδραση με τους χρήστες, σχεδιασμένες με γνώμονα την λειτουργικότητα και την ασφάλεια:

* Command Line Interface (CLI): Αναγνωρίζει την αγγλική γλώσσα, υποστηρίζει εντολές για εισαγωγή δεδομένων, αναζήτηση και προβολή στατιστικών πληροφοριών και αξιοποιεί το REST API για να επικοινωνεί με το backend.
* Web Application (frontend): Λειτουργεί ως client του REST API, επιτρέποντας εύκολη πρόσβαση σε λειτουργίες της εφαρμογής και παρουσιάζει τα δεδομένα σε γραφική μορφή.

Οι δύο αυτές διεπαφές υλοποιούνται με βάση το πρωτόκολλο ασφάλειας HTTPS, για ασφαλή μετάδοση δεδομένων χρησιμοποιώντας self-signed certificates.

# Αναφορές - πηγές πληροφοριών

Αντλήσαμε πληροφορίες από τα πρότυπα SRS που είναι αναρτημένα στην σελίδα του Helios καθώς και από εργαλεία AI και κυρίως το ChatGPT και δευτερεύοντος του Gemini.

# Προδιαγραφές απαιτήσεων λογισμικού

## 3.1 Περιπτώσεις χρήσης

### 3.1.1 ΠΕΡΙΠΤΩΣΗ ΧΡΗΣΗΣ 1: Εμφάνιση χρεών και αναμενόμενων εσόδων του συνδεδεμένου λειτουργού προς και από τους υπολοίπους.

#### 3.1.1.1 Χρήστες (ρόλοι) που εμπλέκονται

* Λειτουργός αυτοκινητόδρομου.
* Η ιστοσελίδα μας.

#### 3.1.1.2 Προϋποθέσεις εκτέλεσης

Οι συνθήκες που πρέπει να ισχύουν προκειμένου να εκτελείται το συγκεκριμένο Use Case είναι οι ακόλουθες :

* Ενημέρωση Βάση Δεδομένων: Όλες οι οικονομικές συναλλαγές, οι οφειλές και τα έσοδα του λειτουργού πρέπει να είναι καταγεγραμμένα και επικαιροποιημένα στη βάση δεδομένων, ώστε να παρέχεται ακριβής και αξιόπιστη πληροφόρηση.
* Τεχνικές προϋποθέσεις: Η εφαρμογή πρέπει να λειτουργεί χωρίς διακοπές και να είναι σε θέση να ανακτά δεδομένα γρήγορα και αξιόπιστα.
* Ο λειτουργός έχει σταθερή και ασφαλής σύνδεση στο ίντερνετ.

#### 3.1.1.3 Περιβάλλον εκτέλεσης

Ο λειτουργός βρίσκεται στην κεντρική σελίδα με τις διαθέσιμες επιλογές. Μόλις επιλέξει το μπουτόν Opetaror Finances θα οδηγηθεί σε μία δεύτερη σελίδα, στην οποία θα του δοθεί η δυνατότητα να επιλέξει ανάμεσα στα μπουτόν Debts Dashboard και Statistics Dashboard. Όταν επιλέξει το πλαίσιο για την εμφάνιση των χρεών θα οδηγηθεί σε σελίδα εκείνη, στην οποία θα μπορέσει να πατήσει τη συντομογραφία του και θα έχει την δυνατότητα να δει όλα τα πραγματικά χρέη του προς τους άλλους (όνομα και οφειλόμενο ποσό). Ενώ όταν επιλέξει το πλαίσιο Statistics θα μπορεί να δει το αναμενόμενο (μελλοντικό) όφελος που θα έχει από τους υπόλοιπους λειτουργούς.

#### 3.1.1.4 Δεδομένα εισόδου

Αφού έχει επιλέξει το πλαίσιο Operator Finances, τα δεδομένα εισόδου περιλαμβάνουν:

* Ο λειτουργός επιλέγει από την κεντρική σελίδα το κουμπί “Debts Dashboard”, πληκτρολόγηση του Tag ID Operator και πάτημα του button “Fetch Debts”
* Ο λειτουργός επιλέγει από την κεντρική σελίδα το κουμπί “Profits Dashboard”, πληκτρολόγηση του Tag ID Operator και πάτημα του button “Fetch Profits”

#### 3.1.1.5 Αλληλουχία ενεργειών - επιθυμητή συμπεριφορά

Βήμα 1: Ο λειτουργός, αρχικά, πατάει τo button “Operator Finances”

Βήμα 2: Έπειτα, έχει δυο δυνατές επιλογές:

1. Αν αποφασίσει να επιλέξει το κουμπί “Debts Dashboard”, θα μπορεί να δει τα χρέη του προς τις υπόλοιπες εταιρίες,
2. Αν αποφασίσει να επιλέξει το κουμπί “Profits Dashboard”, θα μπορεί να δει το μελλοντικό όφελος του από τις υπόλοιπες εταιρίες,

Βήμα 3: Το λογισμικό μας εκτελεί κατάλληλες ενέργειες προκειμένου να ανακτήσει τα ουσιαστικά χρέη του χρήστη (συμψηφισμοί) και κατόπιν αιτήματος από το λογισμικό μας προς το λογισμικό της εταιρίας του λειτουργού, θα στέλνονται σε αυτόν τα χρέη. Αντίστοιχα σε περίπτωση της επιλογής b, το λογισμικό μας θα εκτελεί κατάλληλες ενέργειες προκειμένου να ανακτήσει τα έσοδα που πρόκειται να αποκτήσει ο εκάστοτε λειτουργός. Αν κάτι πάει λάθος, ο χρήστης θα ενημερώνεται με μήνυμα σφάλματος και θα επιστρέφει στο Βήμα 1.

Βήμα 4: Ο λειτουργός είτε ενημερώθηκε για τα χρέη που έχει είτε για τα αναμενόμενα έσοδα, Αν όμως πατήσει “Home”, τότε επανέρχεται στην αρχή της ιστοσελίδας μας στο Bήμα 1.

A diagram of a process

AI-generated content may be incorrect.

State Diagram Use Case 1

A diagram of a website

AI-generated content may be incorrect.

*Activity Diagram Use Case 1*

A diagram of a project

AI-generated content may be incorrect.

Sequence Diagram Use Case 1

#### 3.1.1.7 Δεδομένα εξόδου

Τα δεδομένα εξόδου περιλαμβάνουν την εμφάνιση των χρεών και των μελλοντικών εσόδων του λειτουργού. Σε περίπτωση σφάλματος, λαμβάνει σχετικό μήνυμα στην οθόνη.

### 3.1.2 ΠΕΡΙΠΤΩΣΗ ΧΡΗΣΗΣ 2: Εμφάνιση στατιστικών δεδομένων διελεύσεων από σταθμούς διοδίων.

#### 3.1.2.1 Χρήστες (ρόλοι) που εμπλέκονται

* Ο λειτουργός αυτοκινητόδρομου.
* Η ιστοσελίδα μας.

#### 3.1.2.2 Προϋποθέσεις εκτέλεσης

Οι συνθήκες που πρέπει να ισχύουν προκειμένου να εκτελεστεί η συγκεκριμένη περίπτωση χρήσης είναι οι ακόλουθες:

* Ενημέρωση δεδομένων: Οι καταγεγραμμένες διελεύσεις στη βάση δεδομένων πρέπει να είναι επικαιροποιημένες διασφαλίζοντας την ακρίβεια και την αξιοπιστία των πληροφοριών.
* Άμεση προσβασιμότητα στατιστικών στοιχείων: Τα στατιστικά δεδομένα κάθε σταθμού πρέπει να είναι διαθέσιμα σε πραγματικό χρόνο ή με ελάχιστη καθυστέρηση, επιτρέποντας γρήγορη επεξεργασία και ανάλυση.
* Αντιμετώπιση Ασυνέπειας Δεδομένων: Το σύστημα διαθέτει μηχανισμούς ελέγχου ποιότητας δεδομένων, ώστε να αποφεύγονται διπλές εγγραφές στις καταγραφές διελεύσεων.

#### 3.1.2.3 Περιβάλλον εκτέλεσης

Το περιβάλλον στο οποίο εκτελείται η συγκεκριμένη περίπτωση χρήσης περιλαμβάνει το Web Application (Frontend) της εφαρμογής μας. Τα δεδομένα θα αντλούνται από το REST API του backend, το οποίο θα επικοινωνεί με την σχεσιακή βάση δεδομένων.

Όσον αφορά την Web εφαρμογή, ο λειτουργός, αρχικά, βρίσκεται στην κεντρική σελίδα με τις διαθέσιμες επιλογές. Μόλις επιλέξει το μπουτόν Statistics Dashboard θα οδηγηθεί σε μία δεύτερη σελίδα, στην οποία θα του δοθεί η δυνατότητα πληκτρολογήσει τη συντομογραφία του και να δει όλα τα στατιστικά στοιχεία που έχουν να κάνουν με τις διελεύσεις του από κάθε σταθμό. Συγκεκριμένα, τα στοιχεία αυτά περιλαμβάνουν τη συχνότητα διελεύσεων από κάθε άλλον operator ξεχωριστά και το ποσοστό διέλευσης σε σχέση με όλες τις συνολικές του διελεύσεις από όλους τους λειτουργούς. Παράλληλα με την εμφάνιση αυτή, δίνεται η δυνατότητα αναπαράστασης των ποσοστών σε μορφή πίτας.

#### 3.1.2.4 Δεδομένα εισόδου

Τα δεδομένα εισόδου περιλαμβάνουν:

* Ο λειτουργός επιλέγει από την κεντρική σελίδα το κουμπί “Statistics Dashboard”
* Είσοδο του Tag ID Operator
* Πάτημα του button “Fetch Statistics”

#### 3.1.2.5 Αλληλουχία ενεργειών - επιθυμητή συμπεριφορά

Βήμα 1: Ο λειτουργός θα επιλέγει την ενότητα “Statistics Dashboard” από την κεντρική σελίδα,

Βήμα 2: Ανάλογα τον operator, εκείνος θα πληκτρολογεί τη συντομογραφία της ονομασίας του

Βήμα 3: Το λογισμικό μας θα αποστέλλει αίτημα προς το backend για τα σχετικά στατιστικά δεδομένα μέσω της συνάρτησης fetchStatistics(),

Βήμα 4: α) Το σύστημα θα εμφανίζει τα στατιστικά δεδομένα που ζητήθηκαν αν πρόκειται για επιτυχή αναγνώριση αιτήματος, συγκεκριμένα:

* Σταθμός διέλευσης (**OpID**)
* Συχνότητα διελεύσεων (**frequency**)
* Έσοδα (**profits**)

Τα δεδομένα αυτά θα τα παρουσιάζει σε μορφή πίνακα στο frontend,

β) Σε περίπτωση αποτυχίας ή μη-διαθεσιμότητας των δεδομένων, ενημέρωση του λειτουργού για το πρόβλημα με κατάλληλο μήνυμα,

Βήμα 5: Αρχικοποίηση της ιστοσελίδας ώστε να αναμένει είσοδο από επόμενο λειτουργό.

*A diagram of a process

AI-generated content may be incorrect.*

State Diagram Use Case 2

A diagram of a flowchart

AI-generated content may be incorrect.

Activity Diagram Use Case 2

A diagram of a project

AI-generated content may be incorrect.

Sequence Diagram Use Case 2

#### 3.1.2.7 Δεδομένα εξόδου

Τα δεδομένα εξόδου περιλαμβάνουν κυκλικά διαγράμματα με τα σχετικά στατιστικά δεδομένα σε μορφή πίτας, καθώς και το table-html με τα δεδομένα. Τα δεδομένα περιλαμβάνουν πόσες φορές έχει περάσει ο ίδιος από όλους τους λειτουργούς και το ποσοστό σε σχέση με όλους του τις διελεύσεις.

## 3.2 Απαιτήσεις επιδόσεων

Οι απαιτήσεις επιδόσεων για το συγκεκριμένο use case δεν είναι υψηλές.

Για το πρώτο use case οι απαιτήσεις είναι απαιτητικές μόνο στο δεύτερο κομμάτι της εξόφλησης. Συγκεκριμένα στο πρώτο κομμάτι οι υπολογισμοί των χρεών θα έχουν ήδη γίνει και θα πρέπει μέσα σε 1.5 δευτερόλεπτα να γίνεται η φόρτωση των στοιχείων και μέσα σε 2 δεύτερα να γίνεται η εμφάνιση τους στην οθόνη του λειτουργού. Όμως στην αποπληρωμή οι ταχύτητες πρέπει να είναι πιο γρήγορες. Συγκεκριμένα σε 1 δευτερόλεπτο από την απόφαση εξόφλησης του χρήστη πρέπει να έχει σταλθεί αίτημα πληρωμής στην τράπεζα και από την στιγμή που η τράπεζα στείλει αίτημα επιτυχίας ή αποτυχίας ο χρήστης να έχει ενημερωθεί σε κάτω από 2 δευτερόλεπτα.

Αφού το δεύτερο use case μας δεν απαιτεί υψηλές επιδόσεις και είναι για μόνο μία προβολή στοιχείων πρέπει απλά να γίνεται μέσα σε 1.5 δευτερόλεπτα περίπου η φόρτωση στοιχείων από το database μας(οι υπολογισμοί θα γίνονται συνεχώς αυτόματα κατά την εισαγωγή των δεδομένων στη βάση και κυρίως κατά την εισαγωγή των διελεύσεων) και περίπου πρέπει σε 2 δευτερόλεπτα να γίνεται η απεικόνιση στην οθόνη του χρήστη , που στο use case μας είναι ο συνδεδεμένος λειτουργός.

## 3.3 Απαιτήσεις οργάνωσης δεδομένων

### 3.3.1 Απαιτήσεις και περιορισμοί πρόσβασης σε δεδομένα

Οι συνεργαζόμενοι λειτουργοί θα έχουν πρόσβαση στο σύστημα μέσω διαπιστευτηρίων που θα περιλαμβάνουν μοναδικό όνομα χρήστη και κρυπτογραφημένο κωδικό πρόσβασης. Μέσω αυτών θα μπορούν να έχουν πρόσβαση σε ευαίσθητα οικονομικά δεδομένα, δηλαδή σε αυτά που περιγράφονται από τα use cases μας. Ουσιαστικά λοιπόν είναι απαίτηση στο να υπάρχει στη βάση δεδομένων να υπάρχει ένας πίνακας login που θα αποθηκεύει ονόματα και κωδικούς ώστε να επαληθεύει τους χρήστες και να τους δίνει το δικαίωμα πρόσβασης σε αυτά.

## 3.4 Λοιπές απαιτήσεις

### 3.4.1 Απαιτήσεις διαθεσιμότητας λογισμικού

Η συνεχόμενη λειτουργία των αυτοκινητοδρόμων σε συνδυασμό με την αυξημένη συμφόρηση ειδικά τις ώρες αιχμής στο οδικό δίκτυο της χώρας, καθιστά επιτακτική την ανάγκη το λογισμικό μας να παρέχει υψηλά επίπεδα διαθεσιμότητας και αξιοπιστίας. Συγκεκριμένα, η εφαρμογή οφείλει να παρέχει αδιάλειπτη λειτουργία ώστε να μπορεί να επεξεργάζεται τα αιτήματα των χρηστών και να διαχειρίζεται τα δεδομένα διελεύσεων σε πραγματικό χρόνο. Θα πρέπει, επίσης, να εξασφαλίζει ότι σε περιπτώσεις συντήρησης ή προσωρινής μη-διαθεσιμότητας, θα υπάρχει δυνατότητα αυτόματης αποθήκευσης αιτημάτων για μεταγενέστερη επεξεργασία. Το σύστημα δύναται να παρέχει μηχανισμούς εφεδρικότητας για την αποφυγή απώλειας δεδομένων σε περιπτώσεις αστοχίας καθώς, επίσης, να προάγει την υιοθέτηση μηχανισμών για την αποφυγή υπερφόρτωσης του κεντρικού διακομιστή. Τέλος, θα πρέπει να υποστηρίζει μηχανισμούς παρακολούθησης (monitoring) με τους οποίους θα ανιχνεύει ανωμαλίες και προβλήματα και θα ενημερώνει τους διαχειριστές για τα προβλήματα που επηρεάζουν την διαθεσιμότητα.

Η υποστήριξη της εφαρμογής από υψηλής απόδοσης υπολογιστικές υποδομές είναι κρίσιμη προκειμένου να διασφαλιστούν οι παραπάνω απαιτήσεις.

### 3.4.2 Απαιτήσεις ασφάλειας

Το ζήτημα της ασφάλειας είναι ένα αρκετά ευαίσθητο θέμα και οφείλουμε να διασφαλίσουμε το απόρρητο κάθε χρήστη του λογισμικού μας. Συγκεκριμένα, η σχεδίαση και η υλοποίηση του συστήματος θα πρέπει να διασφαλίζει τα εξής:

* Ιδιωτικότητα δεδομένων

Τα γεγονότα διελεύσεων και οι διακανονισμοί οφειλών θα πρέπει να είναι προσβάσιμα μόνο από τις άμεσα εμπλεκόμενες εταιρείες. Η πρόσβαση να ελέγχεται με βάση το μοντέλο ελάχιστων προνομίων (least privilege), ώστε μόνο οι εξουσιοδοτημένοι χρήστες να βλέπουν τις απαραίτητες πληροφορίες. Θα πρέπει να υπάρχει, επίσης, ελεγχόμενη διαλειτουργικότητα, να μπορούν οι εταιρείες, δηλαδή, να ελέγξουν αποκλειστικά τις οφειλές τους έναντι οποιασδήποτε άλλης εταιρείας, ενώ η πρόσβαση σε δεδομένα άλλων χρηστών ή εταιρειών να αποτρέπεται μέσω ελέγχων ταυτότητας και εξουσιοδότησης. Κάθε έτοιμα θα μπορούσε να καταγράφεται με πλήρη λεπτομέρεια.

* Ασφάλεια δεδομένων

Τα προσωπικά δεδομένα και η εκάστοτε διέλευση κάθε χρήστη από ένα σταθμό διοδίων θα πρέπει να κρυπτογραφείται κατάλληλα προκειμένου να αποφευχθεί η διαρροή των δεδομένων αυτών στο ευρύ κοινό. Όλη η ανταλλαγή δεδομένων μεταξύ των ενδιαφερόμενων να πραγματοποιείται μέσω ασφαλών συνδέσεων SSL/TLS, πιστοποιητικά τα οποία να ανανεώνονται τακτικά. Οι κωδικοί πρόσβασης, επίσης, οφείλουν να αποθηκεύονται με κατάλληλο αλγόριθμο hashing ώστε να διασφαλίζεται η απόκρυψη τους.

* Εποπτεία και διαφάνεια

Θα μπορούσε να γίνεται καταγραφή κάθε πρόσβασης ή τροποποίησης δεδομένων ώστε να διασφαλίζεται η διαφάνεια και να εντοπίζονται πιθανές παραβιάσεις. Το σύστημά μας οφείλει να συμμορφώνεται με τα διεθνή πρότυπα για την προστασία δεδομένων και της διαχείριση πληρωμών. Τέλος, θα μπορούσε οι χρήστες να ενημερώνονται για μη εξουσιοδοτημένες προσπάθειες πρόσβασης ή αλλαγές στα δεδομένα τους.

## 4. Λοιπά UML Διαγράμματα

### 4.1 Deployment Diagram

A diagram of a software server

AI-generated content may be incorrect.

### 4.2 Component Diagram

A diagram of a search engine

AI-generated content may be incorrect.

### 4.3 Class/API Diagram

A diagram of a bus

AI-generated content may be incorrect.

### 4.4 ER Diagram

A screenshot of a computer

AI-generated content may be incorrect.