

ΔΙΑΧΕΙΡΗΣΗ ΕΠΙΧΕΙΡΗΣΙΑΚΩΝ ΔΙΑΔΙΚΑΣΙΩΝ

ΑΠΑΛΛΑΚΤΙΚΗ
ΕΡΓΑΣΙΑ

ΠΕΡΙΕΧΟΜΕΝΑ

Κεφάλαιο 1 : Εισαγωγή	2
<ul style="list-style-type: none"> • Συνοπτική περιγραφή της εργασίας 	
Κεφάλαιο 2 : Μοντέλο Διαδικασίας.....	3
<ul style="list-style-type: none"> • Παρουσίαση του μοντέλου διαδικασίας 	
Κεφάλαιο 3 : Οργανισμός	9
<ul style="list-style-type: none"> • Παρουσίαση του οργανισμού 	
Κεφάλαιο 4 : Μοντέλο Δεδομένων	14
<ul style="list-style-type: none"> • Ανάλυση του μοντέλου δεδομένων 	
Κεφάλαιο 5 : Θέματα Υλοποίησης	22
<ul style="list-style-type: none"> • Αναφορά στα business και process variables που ορίστηκαν, σε επιπλέον στοιχεία που χρησιμοποιήθηκαν στο μοντέλο διαδικασιών, πέραν των tasks και στα φίλτρα ελέγχου πρόσβασης (user filters) που έχουν ενσωματωθεί στη διαδικασία. 	
Κεφάλαιο 6 : Εγχειρίδιο Χρήστη	XX
<ul style="list-style-type: none"> • Λεπτομερές εγχειρίδιο χρήσης με περιγραφή όλων των οθονών του συστήματος. Στο κεφάλαιο αυτό θα πρέπει να περιλαμβάνονται και τα αντίστοιχα screenshots της εφαρμογής. 	

ΚΕΦΑΛΑΙΟ 1: ΕΙΣΑΓΩΓΗ

ΣΥΝΟΠΤΙΚΗ ΠΕΡΙΓΡΑΦΗ ΤΗΣ ΕΡΓΑΣΙΑΣ

Κεντρικό θέμα της εργασίας αποτελεί η δημιουργία ενός διαδικασιοστρεφούς πληροφοριακού συστήματος που θα υλοποιεί την αίτηση και οργάνωση ενός εταιρικού ταξιδιού από κάποιον υπάλληλο της εταιρίας.

Αρχικά δημιουργούμε το μοντέλο διαδικασίας χρησιμοποιώντας τα ανάλογα tasks ώστε να αντικατοπτριστεί πλήρως η εκφώνηση. Χρησιμοποιήθηκαν 2 pools ένα για την κεντρική εταιρία και ένα για το πρακτορείο, θέτοντας τα swim lanes τους ανάλογα με τους συμμετέχοντες στην διαδικασία. Συγκεκριμένα τέθηκαν στο pool της εταιρίας 3 swim lanes ένα για τον υπάλληλο της εταιρίας, ένα για τον manager και ένα για το λογιστή, ενώ στο pool του πρακτορείου τέθηκε 1 swim lane για τον πράκτορα.

Επόμενο βήμα αποτέλεσε η δημιουργία του οργανισμού που θα αποτελείται από δύο groups, με το ένα να αντιστοιχεί στην εταιρία "Company" (με Subgroups Administration και employees) και το άλλο στο πρακτορείο "Travel_Agency" (με Subgroup Agent). Σαν ρόλους θέσαμε 4, ένα για κάθε συμμετέχοντα. Εν τέλει καταχωρήσαμε 4 χρήστες έναν για κάθε ρόλο, ώστε να μπορούμε να συνδεθούμε στην εφαρμογή με την μορφή οποιουδήποτε ρόλου.

Στην συνέχεια δημιουργήσαμε το κατάλληλο μοντέλο δεδομένων προσθέτοντας αντικείμενα και ορίζοντας τα χαρακτηριστικά τους ανάλογα με την εκφώνηση αλλά και βάση της λογικής. Τα αντικείμενα τέθηκαν με βασικό γνώμονα ποιες πληροφορίες επιθυμούμε να αποθηκευτούν κατά την διάρκεια της διαδικασίας.

Για την διεκπεραίωση της επιχειρησιακής διαδικασίας χρειάστηκε η πραγματοποίηση ενεργειών ώστε να υπάρχει σωστή ροή κατά την χρήση του gui. Η σωστή ροή έγινε ρυθμίζοντας την επικοινωνία μεταξύ των pools με σκοπό να μπορούν να δέχονται δεδομένα. Επιπλέον ρυθμίστηκε και ο τρόπος χρήσης των Gateways αλλά και των timer με σκοπό να είναι η ροή η προβλεπόμενη βάση της εκφώνησης. Αυτά σε συνδυασμό με τα process και business variables επεξηγούνται αναλυτικά στο 5ο κεφάλαιο.

Στο τελευταίο κομμάτι της εργασίας συντάχθηκε ένα εγχειρίδιο για την χρήση του συστήματος με σκοπό να είναι πλήρως κατανοητός ο τρόπος λειτουργίας του. Ο χρήστης μέσω της παροχής εικόνων αλλά και μέσω της αναλυτικής περιγραφής για την ακολουθία της επιχειρηματικής διαδικασίας και τον τρόπο συμπλήρωσης των φορμών, θα είναι σε θέση να χρησιμοποιήσει ορθά το σύστημα.

ΚΕΦΑΛΑΙΟ 2: ΜΟΝΤΕΛΟ ΔΙΑΔΙΚΑΣΙΑΣ

ΠΑΡΟΥΣΙΑΣΗ ΤΟΥ ΜΟΝΤΕΛΟΥ ΔΙΑΔΙΚΑΣΙΑΣ

Ανάλυση Ρόλων:

- Υπάλληλος εταιρίας: Αρχικά ο υπάλληλος οφείλει να αποστείλει στην διοίκηση της εταιρίας μια αίτηση διεκπεραίωσης εταιρικού ταξιδιού η οποία θα περιέχει τόσο τα δικά του προσωπικά στοιχεία, όσο και τα στοιχεία του ταξιδιού που θέλει να πραγματοποιήσει. Στην περίπτωση που το αίτημα γίνει αποδεκτό από κάποιον υπεύθυνο της εταιρίας, ζητάει από το συνεργαζόμενο ταξιδιωτικό γραφείο να προσφερθεί κάποιο πακέτο ταξιδιού με τα δεδομένα που του έχουν δοθεί. Έπειτα ο ταξιδιωτικός πράκτορας θα του στείλει μία προσφορά την οποία ο υπάλληλος θα πρέπει να αξιολογήσει και να επιλέξει αν θα την αποδεχτεί ή αν θα ζητήσει νέα προσφορά. Σε περίπτωση που ο υπάλληλος αποφασίσει να δεχτεί την υπάρχουσα προσφορά, πρέπει να πληρώσει το αντίστοιχο ποσό και όταν έρθει ή κατάλληλη ημερομηνία να αναχωρήσει για το ταξίδι. Τέλος μετά το πέρας του ταξιδιού ο χρήστης υπολογίζει τα έξοδά του και τα στέλνει στον υπεύθυνο που του ενέκρινε την αίτηση, με σκοπό να λάβει το απαραίτητο ποσό από το λογιστήριο.
- Manager (Μέλος διοίκησης της εταιρίας): Ο υπεύθυνος δέχεται από τον υπάλληλο ένα αίτημα με πληροφορίες για το ταξίδι που επιθυμεί να πραγματοποιήσει και αξιολογεί εάν είναι αποδεκτό το αίτημα ενημερώνοντας ανάλογα τον υπάλληλο. Μετά το πέρας του ταξιδιού θα λάβει από τον υπάλληλο τα έξοδα του ταξιδιού και θα τα στείλει στο λογιστήριο έτσι ώστε να εξοφληθεί ο υπάλληλος.
- Λογιστής (Λογιστήριο): Ο λογιστής έχει σαν μοναδική δουλειά να αποστείλει στον υπάλληλο της εταιρίας τα χρήματα που προέκυψαν από το ταξίδι.
- Πράκτορας (Ταξιδιωτικό γραφείο): Ο πράκτορας θα λάβει το αίτημα από τον υπάλληλο με τις πληροφορίες του ταξιδιού θα το καταγράψει και μετά από έρευνα θα στείλει μια προσφορά στον ενδιαφερόμενο. Έχει την υποχρέωση σε περίπτωση που ο χρήστης δεν απαντήσει εντός του χρονικού ορίου των 5 ημερών να τον ειδοποιεί έως ότου απαντήσει. Σε περίπτωση ασυμφωνίας για την αποσταλμένη προσφορά ο πράκτορας θα αποστείλει άλλη προσφορά. Όταν καταλήξουν σε συμφωνία θα πραγματοποιήσει την κράτηση κανονίζοντας παράλληλα την ασφάλεια του ταξιδιού. Στην συνέχεια θα ενημερώσει τον υπάλληλο για το συνολικό

ποσό και θα περιμένει για την πληρωμή. Σε περίπτωση μη καταβολής του ποσού θα τερματίσει την κράτηση, ενώ σε αντίθετη περίπτωση θα δώσει στον πελάτη (υπάλληλο εταιρίας) λεπτομερείς πληροφορίες για το ταξίδι του.

Ανάλυση Actors:

Οι actors είναι έμμεσες κατευθύνσεις μεταξύ των διαδικασιών και των οντοτήτων του οργανισμού. Χρησιμοποιούνται για τον καθορισμό των υποψηφίων που μπορούν να εκτελέσουν ένα δεδομένο human task. Έτσι έχουν οριστεί actors για κάθε lane του διαγράμματος μας.

Συγκεκριμένα στο Employee Lane έχει οριστεί ως actor ο Employee καθώς στο συγκεκριμένο lane βρίσκονται human tasks τα οποία μπορεί να κάνει μόνο ένας employee της εταιρίας. Επίσης στο Pool του Company έχουμε δώσει στον Employee Actor την ιδιότητα του initiator καθώς αυτός είναι ο μόνος που μπορεί να ξεκινήσει την όλη διαδικασία.

Στο Administration lane έχουμε ορίσει ως actor τον manager καθώς στο συγκεκριμένο lane περιλαμβάνονται tasks τα οποία μπορούν να πραγματοποιήσουν μόνο άτομα της εταιρίας με την ιδιότητα του manager.

Επίσης στο Finance Department έχουμε ορίσει ως actor τον accountant αφού μόνο άτομα της εταιρίας με την ιδιότητα του λογιστή μπορούν να κάνουν μετακινήσεις χρημάτων από τα ταμεία της εταιρίας προς τους υπαλλήλους της.

Τέλος για το Travel Agent lane έχουμε ορίσει ως actor τον travel agent καθώς μόνο αυτός μπορεί να ολοκληρώσει τα human task που βρίσκονται μέσα στο συγκεκριμένο lane. Στο pool του Travel Agency δεν έχει οριστεί κάποιος actor ως initiator καθώς δεν μπορεί κάποιος από το ταξιδιωτικό γραφείο να εκκινήσει ολόκληρη τη διαδικασία.

Ανάλυση Ροής:

- Ο υπάλληλος της εταιρίας πραγματοποιεί αίτημα για να πραγματοποιήσει ένα ταξίδι (Human Task-->Issue Travel Request)
- Ο manager αξιολογεί εάν είναι αποδεκτό το αίτημα ταξιδιού (Human Task -->Review Travel Request)
- Έπειτα υπάρχει μια πύλη XOR Split που ελέγχει εάν το αίτημα έγινε αποδεκτό και σπάει σε δύο μονοπάτια που θα ακολουθηθεί μόνο το ένα (XOR Split --> Is Request Accepted?)
 - Εάν δεν έγινε αποδεκτό το αίτημα ειδοποιείται ο υπάλληλος (Human Task --> Notify Employee For Request Denial)
 - Τερματίζεται η διαδικασία(End 5)
- Ο υπάλληλος δημιουργεί μια φόρμα σχετικά με το ταξίδι που θέλει να πραγματοποιήσει(Human Task --> Form Offer Request To Agent)
- Με την χρήση Throw Message στέλνεται προς τον πράκτορα η φόρμα (Throw Message --> Message 1)
- Με την χρήση Start Message λαμβάνει το μήνυμά ο πράκτορας (Start Message --> Start2)
- Δημιουργία μιας Business Variable από το Persistence Id, που έλαβε ο πράκτορας, σαν Service Task (Service Task --> Build Business Var From Persistence Id Received)
- Ο πράκτορας καταγράφει το αίτημα για ταξίδι (Human Task --> Record Request)
- Ο πράκτορας ψάχνει για να βρει προσφορά που αντικατοπτρίζει τα χαρακτηριστικά του αιτήματος (Human Task --> Search for Initial Offer)
- Αποστολή προσφοράς από τον πράκτορα στον υπάλληλο με την χρήση Throw Message (Throw Message --> Send Initial Offer)
 - Το Throw Message συνδέεται επίσης με XOR Join που σε αυτό συνδέεται επίσης και το Throw Message της αποστολής εναλλακτικής προσφοράς(Αναλύεται παρακάτω στην ροή) (XOR Split --> Which Message Was Sent?)
- Ο υπάλληλος λαμβάνει την προσφορά με το πακέτο ταξιδιού από τον πράκτορα με την χρήση Catch Message (Catch Message --> Receive Initial Offer)
- Δημιουργία Business Variable από το Persistence Id, που έλαβε ο υπάλληλος, με Service Task (Service Task --> Build Business Var From Persistence Id Received)
- Στην συνέχεια υπάρχει ένα XOR Join που σε αυτό ενώνονται η αξιολόγηση της υπενθύμισης από τον υπάλληλο(Πιο κάτω στην ροή), το Service Task για την δημιουργία του Business Variable και το Service Task που αντιγράφει την δημιουργία του Business Variable(Κατά την διαδικασία που ο πράκτορας στέλνει εναλλακτική προσφορά). (XOR Join --> Which Message Was Received?)

- Ο υπάλληλος της εταιρίας αξιολογεί την προσφορά για το ταξίδι που έστειλε ο πράκτορας (Human Task --> Review Trip Offer). Σε αυτό το Human task έχουμε θέσει ένα timer με διάρκεια 5 ημερών (Timer --> 5-Day Duration).
 - Στην περίπτωση που το task δεν ολοκληρωθεί εντός 5 ημερών τότε η διαδικασία οδηγείται με την χρήση του timer στο εναλλακτικό μονοπάτι που πρέπει να ειδοποιηθεί ο υπάλληλος για να δώσει απάντηση
 - Με την χρήση Throw Message στέλνεται μήνυμά προς τον πράκτορα ότι δεν έχει απαντήσει ο υπάλληλος (Throw Message --> Signal1)
 - Ο πράκτορας λαμβάνει το μήνυμα ότι δεν έστειλε ο πράκτορας απάντηση για την προσφορά με την χρήση Start Message (Start Message --> Message4)
 - Ο πράκτορας δημιουργεί επιστολή υπενθύμισης για να την στείλει στον υπάλληλο (Human Task --> Create Reminder)
 - Με την χρήση End Message στέλνει υπενθύμιση προς τον υπάλληλο (End Message --> Message5)
 - Ο υπάλληλος δέχεται την υπενθύμιση με την χρήση Catch Message (Catch Message --> Receive Reminder)
 - Δημιουργία αντιγράφου Business Variable από το Persistence Id που λήφθηκε, με την χρήση Service Task (Service Task --> Build Business Var From Persistence Id Received 3)
 - Ο υπάλληλος αξιολογεί την υπενθύμιση που του στάλθηκε (Human Task --> Review Reminder)
 - Φτάνει πάλι στην κατάσταση που περνάει από το XOR Join (XOR Join --> Which Message Was Received?)
- Εάν το Human Task (Review Trip Offer) ολοκληρωθεί πριν ο timer τελειώσει τότε θα στέλνεται ένα μήνυμά προς τον πράκτορα με την απάντηση. Χρησιμοποιούμε το Throw Message για την αποστολή μηνύματος (Throw Message --> Message3)
- Με την χρήση Catch Message ο πράκτορας λαμβάνει το μήνυμά του υπαλλήλου (Catch Message --> Message2)
- Δημιουργία Business Variable από το Persistence Id, που έλαβε ο πράκτορας, με Service Task (Service Task --> Build Business Var From Persistence Id Received 2)
- Μετά υπάρχει ένα XOR Split που διαλέγει μονοπάτι ανάλογα με το αν ο υπάλληλος ανέφερε εάν αποδέχεται την προσφορά (XOR Split --> Is Offer Accepted?)
 - Εάν δεν έχει αποδεχθεί την προσφορά ο υπάλληλος, τότε ο πράκτορας ψάχνει εναλλακτικές προσφορές για να προτείνει στον υπάλληλο (Human Task --> Search for Alternative Offers)
 - Έπειτα με την χρήση μηνύματος Throw Message στέλνεται η

- προσφορά από τον πράκτορα στον υπάλληλο (Throw Message --> Send Alternative Offer)
- Παράλληλα πηγαίνει σε ένα XOR Split (Which Message Was Sent?) που σε αυτό συνδέεται επίσης η αποστολή της αρχικής προσφοράς
- Ο υπάλληλος δέχεται την εναλλακτική προσφορά του πράκτορα με την χρήση Catch Message (Catch Message --> Receive Alternative Offer)
- Στην συνέχεια με την χρήση ενός Service Task αντιγράφεται το Business Variable από το Persistence Id που παραδόθηκε (Service Task --> Copy Of Build Business Var From Persistence Id Received)
- Τέλος οδηγείται στο XOR Split Which Message Was Received? που αυτό θα συνεχίσει με τον έλεγχο της προσφοράς από τον υπάλληλο, μπαίνοντας δηλαδή ξανά την κανονική ροή
- Εάν αποδεχθεί την προσφορά τότε πηγαίνει η ροή σε ένα AND Split όπου θα σπάσει σε δύο διαδικασίες (AND Split --> After Acceptance1). Πρέπει να ολοκληρωθούν και οι δύο διαδικασίες για να συνεχίσει η διαδικασία.
 - Γίνεται η κράτηση του ταξιδιού με την χρήση Service Task (Service Task --> Make Reservation)
 - Ο πράκτορας κανονίζει για την ασφάλεια ταξιδιού που θα παρέχεται στον υπάλληλο (Human Task --> Arrange Trip Insurance)
- Μετά υπάρχει μια AND Join που εκεί καταλήγουν τα Task Make Reservation και Arrange Trip Insurance. Πρέπει να ολοκληρωθούν και τα 2 ώστε να συνεχίσει η διαδικασία μετά την πύλη (AND Join --> After Acceptance2)
- Ο πράκτορας δημιουργεί τον λογαριασμό της προσφοράς που αποδέχθηκε ο υπάλληλος (Human Task --> Create Bill)
- Με την χρήση Throw Message θα σταλθεί ο λογαριασμός στον πελάτη (Throw Message --> Send Bill)
- Παράλληλα ενώνεται το Throw message με ένα Catch Message που περιμένει να δεχθεί την πληρωμή (Αναλύεται αργότερα στην ροή)
- Ο υπάλληλος δέχεται τον λογαριασμό με την χρήση ενός Catch Message (Throw Message --> Receive Bill)
- Build Business Var From Persistence Id Received 4
- Ο υπάλληλος εκτελεί την διαδικασία της πληρωμής και είτε την ολοκληρώνει είτε όχι (Human Task --> Pay Bill)
- Στέλνεται το αποτέλεσμα της πληρωμής στον πράκτορα με την χρήση Throw Message (Throw Message --> Receive Payment)
- Ο πράκτορας ελέγχει εάν η πληρωμή έχει πραγματοποιηθεί (Human Task --> Check If payment is completed)

- Μετά η ροή οδηγείται σε ένα XOR Split που θα ελέγχει εάν η πληρωμή έχει πραγματοποιηθεί από την μεριά του πελάτη και θα ακολουθείται το ανάλογο μονοπάτι(XOR Split --> Is Payment Complete)
 - Εάν δεν έχει ολοκληρωθεί η πληρωμή τότε ακυρώνεται αυτόματα με Service Task η κράτηση (Cancel Reservation)
 - Τερματίζεται η διαδικασία(End 2)
- Εάν έχει πραγματοποιηθεί η πληρωμή τότε ο πράκτορας καταγράφει τις λεπτομερείς πληροφορίες για το ταξίδι (Human Task --> Create Trip Details)
- Οι λεπτομερείς πληροφορίες του ταξιδιού στέλνονται στον υπάλληλο με την χρήση Throw Message(Throw Message --> Message6)
- Ο υπάλληλος δέχεται τις λεπτομερείς πληροφορίες του ταξιδιού με την χρήση Catch Message(Catch Message --> Message7)
- Build Business Var From Persistence Id Received 5
- Ο υπάλληλος μελετά και αξιολογεί τις πληροφορίες του ταξιδιού (Human Task --> Review Trip Details)
- Ο υπάλληλος ολοκληρώνει το ταξίδι(Human Task --> Complete Trip)
- Γίνεται έπειτα χρήση timer για να περάσει μια μέρα (ώστε μετά να στείλει τα έξοδα)(Timer --> A Day After Trip)
- Ο υπάλληλος δημιουργεί μια φόρμα που σε αυτή καταγράφει όλες τις δαπάνες χρημάτων που πραγματοποίησε (Human Task --> Create Expenses Form)
- Ο manager διερευνά τις συνολικές δαπάνες και ενημερώνει το λογιστήριο για πίστωση χρημάτων (Human Task --> Examine Expenses)
- Έπειτα ο λογιστής συμπληρώνει τα προσωπικά του δεδομένα (Human Task --> Send Payment)
- Ο υπάλληλος λαμβάνει τα χρήματα που δικαιούται από το λογιστήριο(Human Task --> Receive Payment)
- Τερματίζεται η διαδικασία(End4)

ΚΕΦΑΛΑΙΟ 3 : ΟΡΓΑΝΙΣΜΟΣ

ΠΑΡΟΥΣΙΑΣΗ ΤΟΥ ΟΡΓΑΝΙΣΜΟΥ

ΑΝΑΛΥΣΗ

Organization Groups: Σε αυτό το σημείο της εργασίας θα θέσουμε δυο organisation groups με γνώμονα ότι έχουμε δυο pools.



- 1) Το ένα group θα είναι το “Company” που θα αναπαριστά το κομμάτι της εταιρίας. Τα subgroups αυτουνού θα είναι 2.

Company

Name
Company

Display name
Company

Path
Company

Path is required to use groups through API's.

Description

- Έχουμε το “Employees” που αναφέρεται στους υπαλλήλους της εταιρίας που έχουν ως σκοπό την πραγματοποίηση ενός ταξιδιού, αλλά και στους λογιστές της εταιρίας που έχουν ως μοναδική αρμοδιότητα την καταβολή χρημάτων.

Employees

Name
Employees

Display name
Employees


Path
/Company/Employees

Path is required to use groups through API's.


Description

- Έχουμε το "Administration" που αναφέρετε στους managers της εταιρίας. Ένας manager θα έχει τα διαπλεκόμενα στο ταξίδι του υπαλλήλου.

Administration

Name
Administration 

Display name
Administration


Path
/Company/Administration 

Path is required to use groups through API's.


Description

- 2) Το άλλο group θα είναι το "Travel_Agency" που θα αναπαριστά το κομμάτι του ταξιδιωτικού πρακτορείου. Το subgroup αυτού θα είναι μόνο 1.

Travel_Agency

Name
Travel_Agency 

Display name
Travel_Agency


Path
Travel_Agency 

Path is required to use groups through API's.


Description

- Έχουμε το "Agent" που αναπαριστά τον πράκτορα που εργάζεται στο "Travel_Agency" και συνεννοείται με τον πελάτη για τις λεπτομέρειες του ταξιδιού.

Agent

Name
Agent 

Display name
Agent


Path
/Travel_Agency/Agent 

Path is required to use groups through API's.

Description

Organization roles: Σε αυτό το σημείο της εργασίας θα θέσουμε 4 roles που αναπαριστούν καθένα από τον συμμετέχοντα σε αυτή την επιχειρησιακή διαδικασία. Με άλλα λόγια όσα lanes έχουμε τόσα θα είναι και τα roles.

Organization roles

 Company Employees
 Travel Agent
 Company Managers
 Company Accountants

1) Company Employees: Υπάλληλοι της εταιρίας

Company Employees

Name

Display name

Description

2) Travel Agent: Πράκτορες ταξιδιωτικού πρακτορείου

Travel Agent

Name

Display name

Description

3) Company Managers: Managers της εταιρίας

Company Managers

Name

Display name

Description

4) Company Accountants: Λογιστές της εταιρίας

Company Accountants

Name

Company Accountants

Display name

Company Accountants

Description

Organization users: Σε αυτό το κομμάτι θα θέσουμε τους χρήστες που θα έχει το organisation. Έχουμε θέσει στον καθένα ένα Username και ένα Password που μέσω αυτών θα έχουν το δικαίωμα πρόσβασης στην εταιρία. Επιπλέον ο καθένας από αυτούς έχει ένα συγκεκριμένο ρόλο στην εταιρία. Έχουμε θέσει 4 ρόλους οπότε έχουμε δημιουργήσει 4 users έναν για κάθε ρόλο.

Organization users

Eric Gige

Eva Old

John Coe

Mac Kots

- Υπάρχει ένας με ρόλο "Company Accountant" που ανήκει στο Subgroup "Employees" του Group "Company". Ονομάζεται Mac Kots και έχει Username: accountant και Password: mac. Έχει σαν manager τον John Coe.

Mac Kots

Username

accountant

Password

mac

First name

Mac

Last name

Kots

Title

Mr

Job title

Accountant

Manager

manager

Memberships

+ Add - Delete

Group	Role
/Company/Employees	Company Accountants

- Υπάρχει ένας με ρόλο "Company Employees" που ανήκει στο Subgroup "Employees" του Group "Company". Ονομάζεται Eric Gige και έχει Username: employee και Password: eric. Έχει σαν manager τον John Coe

Eric Gige

Username
employee

Password
eric

First name
Eric

Last name
Gige

Title
Mr, Ms...

Job title
HR manager, Junior Sales...

Manager
manager

Memberships

+ Add - Delete

Group	Role
/Company/Employees	Company Employees

- Υπάρχει μια με ρόλο "Travel Agent" που ανήκει στο Subgroup "Agent" του Group "Travel Agency". Ονομάζεται Eva Old και έχει Username: TravelAgent και Password: eva

Eva Old

Username
TravelAgent

Password
eva

First name
Eva

Last name
Old

Title
Mr, Ms...

Job title
HR manager, Junior Sales...

Manager

Memberships

+ Add - Delete

Group	Role
/Travel_Agency/Agent	Travel Agent

- Υπάρχει ένας με ρόλο "Company Mangers" που ανήκει στο Subgroup "Administration" του Group "Company". Ονομάζεται John Coe και έχει Username: manager και Password: john.

John Coe

Username
manager

Password
john

First name
John

Last name
Coe

Title
Mr, Ms...

Job title
HR manager, Junior Sales...

Manager

Memberships

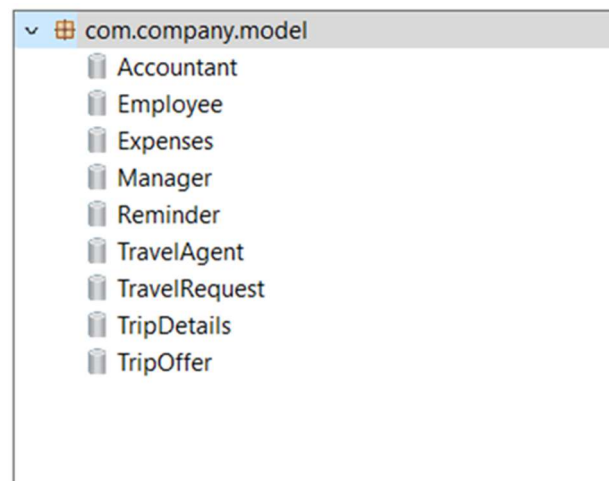
+ Add - Delete

Group	Role
/Company/Administration	Company Managers

ΚΕΦΑΛΑΙΟ 4 : ΜΟΝΤΕΛΟ ΔΕΔΟΜΕΝΩΝ

ΠΑΡΟΥΣΙΑΣΗ ΤΟΥ ΜΟΝΤΕΛΟΥ ΔΕΔΟΜΕΝΩΝ

Business Data Model: Σε αυτό το section θα δημιουργήσουμε Object βάση των δεδομένων που θέλουμε να αποθηκεύουμε στην επιχειρησιακή διαδικασία. Σε κάθε object θα θέσω τα απαραίτητα στοιχεία βάση της λογικής αλλά και βάση των αναγκών που προκύπτουν στην εργασία. Τα δεδομένα αυτά είναι τα παρακάτω



Objects:

Accountant: Σε αυτό το object θα αποθηκεύσουμε πληροφορίες που έχουν σχέση με τον λογιστή.

Χαρακτηριστικά:

- accountantName --> Όνομα λογιστή (τύπου String)
- accountantSurname --> Επώνυμο λογιστή (τύπου String)
- accountantId --> Μοναδικό αναγνωριστικό λογιστή (τύπου String)
- accountantEmail --> Ηλεκτρονικό ταχυδρομείο λογιστή (τύπου Text)
- accountantPhoneNumber --> Αριθμός κινητού τηλεφώνου λογιστή (τύπου Long)
- accountantHomeNumber --> Αριθμός σταθερού τηλεφώνου λογιστή (τύπου Long)
- accountantFax --> Αριθμός fax του λογιστή (τύπου Long)
- accountantAddressStreet --> Οδός κατοικίας λογιστή (τύπου String)
- accountantAddressNumber --> Αριθμός οδού κατοικίας λογιστή (τύπου Integer)

Από αυτά τα χαρακτηριστικά έχω θέσει τρία σαν Mandatory (δηλαδή υποχρεωτικά) και συγκεκριμένα είναι τα

- 1)accountantEmail
- 2)accountantPhoneNumber
- 3)accountantId

Employee: Σε αυτό το object θα αποθηκεύσουμε πληροφορίες που έχουν σχέση με τους υπαλλήλους της εταιρίας.

Χαρακτηριστικά:

- employeeId --> Μοναδικό διακριτικό υπαλλήλου (τύπου String)
- name --> Όνομα υπαλλήλου (τύπου String)
- surname --> Επώνυμο υπαλλήλου (τύπου String)
- jobTitle --> Θέση στην δουλειά (τύπου String)
- department --> Τμήμα που ανοίκει στην εταιρία (τύπου String)
- email --> Ηλεκτρονικό ταχυδρομείο υπαλλήλου (τύπου String)
- phoneNumber --> Αριθμός κινητού τηλεφώνου του υπαλλήλου (τύπου Long)
- homePhoneNumber --> Αριθμός σταθερού τηλεφώνου του υπαλλήλου (τύπου Long)
- fax --> Αριθμός Fax του υπαλλήλου (τύπου Long)
- addressStreet --> Όνομα οδού κατοικίας του υπαλλήλου (τύπου String)
- addressNumber --> Αριθμός οδού κατοικίας του υπαλλήλου

Από αυτά τα χαρακτηριστικά έχουμε θέσει τρία σαν Mandatory (δηλαδή υποχρεωτικά) και συγκεκριμένα είναι τα

- 1)employeeId
- 2)email
- 3)phoneNumber

Expenses: Σε αυτό το object θα αποθηκεύσω πληροφορίες που έχουν σχέση με τα έξοδα που έκαναν οι υπάλληλοι για την διεκπεραίωση του ταξιδιού.

Χαρακτηριστικά:

- extraHotelCost --> Συνολικό κόστος ξενοδοχείου (τύπου Long)

- `extraTransitCost` --> Συνολικό κόστος μεταφορών (τύπου Long)
- `extraAirplaneCost` --> Συνολικό κόστος αεροπλάνου (τύπου Long)
- `coveredByAdministrationCost` --> Χρήματα που καλύπτονται από την διοίκηση (τύπου Long)
- `finalCost` --> Τελικό κόστος (τύπου Long)
- `areExpensesCovered` --> Boolean για διεκπεραίωση εάν οι δαπάνες θα καλυφθούν (τύπου Boolean)
- `areExpensesCorrect` --> Boolean για διεκπεραίωση εάν οι δαπάνες που καταγράφηκαν είναι σωστές (τύπου Boolean)
- `accountantInCharge` --> Λογιστής που είναι υπεύθυνος για την αποστολή χρημάτων στον υπάλληλο (τύπου Accountant)

Από τα χαρακτηριστικά που έχω θέσει δεν είναι κανένα mandatory. Το `accountantInCharge` είναι τύπου του `Object Accountant` και θα θέσουμε σαν relationship "Aggregation".

Manager: Σε αυτό το object θα αποθηκεύσω πληροφορίες που έχουν σχέση με τους managers της εταιρίας. Χαρακτηριστικά:

- `managerSurname` --> Επώνυμο του manager (τύπου String)
- `managerName` --> Όνομα του manager (τύπου String)
- `managerID` --> Μοναδικό διακριτικό manager (τύπου String)
- `managerDepartment` --> Τμήμα που ανοίκει ο manager στην εταιρία (τύπου String)
- `managerJobTitle` --> Τίτλος θέσης του manager στην εταιρία (τύπου String)
- `managerEmail` --> Ηλεκτρονικό ταχυδρομείο του manager (τύπου String)
- `managerPhoneNumber` --> Αριθμός κινητού τηλεφώνου manager (τύπου Long)
- `managerHomeNumber` --> Αριθμός σταθερού τηλεφώνου manager (τύπου Long)
- `managerAddressStreet` --> Όνομα οδού κατοικίας manager (τύπου String)
- `managerAddressNumber` --> Αριθμός οδού κατοικίας manager (τύπου Integer)

- `authorizedmanager` --> Boolean για την διεκπεραίωση εάν είναι ο Manager, που είναι υπεύθυνος για τον έλεγχο του υπαλλήλου. (τύπου Boolean)

Από αυτά τα χαρακτηριστικά έχω θέσει τρία σαν Mandatory (δηλαδή υποχρεωτικά) και συγκεκριμένα είναι τα

- 1) `managerEmail`
- 2) `managerPhoneNumber`
- 3) `managerID`

Reminder: Σε αυτό το object θα θέσω μόνο ένα Boolean που θα το χρησιμοποιήσουμε σε XOR για να στέλνουμε μήνυμά από τον πράκτορα στον υπάλληλο, με σκοπό να υπενθυμίσει ο πρώτος για την προσφορά που έκανε ώστε ο δεύτερος να δώσει τελική απάντηση εάν τον ενδιαφέρει η προσφορά ή όχι.

- `isClientStillInterested`(τύπου Boolean)

Travel Agent: Σε αυτό το Object θα αποθηκεύσω πληροφορίες που έχουν σχέση με τους Travel Agents της εταιρίας.

Χαρακτηριστικά:

- `taName` --> Όνομα του πράκτορα (τύπου String)
- `taSurname` --> Επώνυμο του πράκτορα (τύπου String)
- `taEmail` --> Ηλεκτρονικό Ταχυδρομείο του πράκτορα (τύπου String)
- `taPhoneNumber` --> Αριθμός κινητού τηλεφώνου πράκτορα (τύπου Integer)
- `taFax` --> Αριθμός fax του πράκτορα (τύπου Long)
- `taAddressStreet` --> Όνομα οδού κατοικίας του πράκτορα (τύπου String)
- `taAddressNumber` --> Αριθμός οδού κατοικίας του πράκτορα (τύπου Integer)

Από αυτά τα χαρακτηριστικά έχουμε θέσει τέσσερα σαν Mandatory(δηλαδή υποχρεωτικά) και συγκεκριμένα είναι τα

- 1) `taName`
- 2) `taSurname`
- 3) `taEmail`
- 4) `taPhoneNumber`

TravelRequest: Σε αυτό το Object θα αποθηκεύσω πληροφορίες που έχουν σχέση με τα Travel Requests που γίνονται από τους υπαλλήλους.
Χαρακτηριστικά:

- `applier` --> Ο υπάλληλος που κάνει την αίτηση του ταξιδιού (τύπου `Employee`)
- `leavesOn` --> Ημερομηνία αναχώρησης από το ταξίδι (τύπου `Datetime`)
- `returnsOn` --> Ημερομηνία επιστροφής από το ταξίδι (τύπου `Datetime`)
- `country` --> Χώρα που θα πάει ο υπάλληλος (τύπου `String`)
- `city` --> Πόλη που θα πάει ο υπάλληλος (τύπου `String`)
- `travelReason` --> Λόγος πραγματοποίησης του ταξιδιού (τύπου `String`)
- `isRequestAccepted` --> Boolean για το αν το αίτημα του ταξιδιού έχει εγκριθεί από τον υπεύθυνο `manager` (τύπου `Boolean`)
- `supervisedByManager` --> `Manager` που θα εγκρίνει το αίτημα για το ταξίδι (τύπου `Manager`)

Από αυτά τα χαρακτηριστικά έχω θέσει έξι σαν Mandatory (δηλαδή υποχρεωτικά) και συγκεκριμένα είναι τα

- 1) `applier`
- 2) `leavesOn`
- 3) `returnsOn`
- 4) `country`
- 5) `city`
- 6) `travelReason`

Το χαρακτηριστικό `applier` το έχουμε θέσει με τύπο `Employee(Object)` και `Relationship "Composition"`

Το χαρακτηριστικό `supervisedByManager` το έχουμε θέσει με τύπο `Manager(Object)` και `Relationship "Composition"`

Trip Details: Σε αυτό το Object θα αποθηκεύσω τις λεπτομέρεις πληροφοριών του κάθε ταξιδιού.

Χαρακτηριστικά:

- `defAirline` --> Εταιρία αεροπλάνου (τύπου `String`)
- `defElectronicTicketNumber` --> Αριθμός ηλεκτρονικού εισιτηρίου (τύπου `Integer`)
- `gate` --> Πύλη αναχώρησης αεροπλάνου (τύπου `Integer`)
- `seat` --> Θέση στο αεροπλάνο (τύπου `String`)
- `defDeparture` --> Ημερομηνία αναχώρησης του αεροπλάνου (Προς προορισμό) (τύπου `Datetime`)

- `detArrival` --> Ημερομηνία άφιξης του αεροπλάνου (Προς προορισμό)(τύπου `Datetime`)
- `fromAirport` --> Αεροδρόμιο αναχώρησης (τύπου `String`)
- `toAirport` --> Αεροδρόμιο άφιξης(τύπου `String`)
- `flightId` --> Μοναδικό διακριτικό πτήσης αναχώρησης (τύπου `String`)
- `returnFlightId` --> Μοναδικό διακριτικό πτήσης επιστροφής (τύπου `String`)
- `detReturnDeparture` --> Ημερομηνία αναχώρησης του αεροπλάνου(Προς κατοικία)(τύπου `Datetime`)
- `detReturnArrival` --> Ημερομηνία άφιξης του αεροπλάνου(Προς κατοικία) (τύπου `Datetime`)
- `returnFromAirport` --> Αεροδρόμιο επιβίβασης για επιστροφή(Αεροδρόμιο προορισμού που έχει ταξιδέψει ο υπάλληλος) (τύπου `String`)
- `returnToAirport` --> Αεροδρόμιο αποβίβασης κατά την επιστροφή(Αεροδρόμιο κατοικίας του υπαλλήλου) (τύπου `String`)
- `returnGate` --> Πύλη αεροδρομίου κατά την επιστροφή (τύπου `Integer`)
- `returnAirline` --> Εταιρία αεροπλάνου κατά την επιστροφή (τύπου `String`)
- `returnSeat` --> Θέση στο αεροπλάνο κατά την επιστροφή (τύπου `String`)
- `transitSpotPickUp` --> Διεύθυνση από όπου θα παραλάβουν τον υπάλληλο στην τοποθεσία που ταξίδεψε (τύπου `String`)
- `returnTransitSpotPickUp` --> Διεύθυνση από όπου θα παραλάβουν τον υπάλληλο κατά την επιστροφή (τύπου `String`)
- `detHotelName` --> Όνομα ξενοδοχείου διαμονής (τύπου `String`)
- `detHotelAddressStreet` --> Όνομα οδού που βρίσκεται το ξενοδοχείο (τύπου `String`)
- `detHotelAddressNumber` --> Αριθμός οδού που βρίσκεται το ξενοδοχείο (τύπου `Integer`)
- `detHotelAmenities` --> Υπηρεσίες που παρέχει το ξενοδοχείο (τύπου `String`)
- `detHotelTypeOfRoom` --> Τύπος δωματίου υπαλλήλου στο ξενοδοχείο (τύπου `String`)
- `detTripInsurance` --> `Boolean` για επιλογή εάν θα υπάρχει ασφάλεια ταξιδιού που παρέχεται στον υπάλληλο (τύπου `Boolean`)

- agreementTermsAndConditions --> Boolean για να επιλέξει εάν δέχεται του όρους και τις προϋποθέσεις (τύπου Boolean)

Από αυτά τα χαρακτηριστικά έχω θέσει έξι σαν Mandatory(δηλαδή υποχρεωτικά) και συγκεκριμένα είναι τα

- 1)flightId
- 2)returnFlightId
- 3)transitSpotPickup
- 4)returnTransitSpotPickUp
- 5)detHotelName
- 6)detHotelTypeOfRoom
- 7)agreementTermsAndCondition

Επιπλέον σε αυτά τα χαρακτηριστικά έχω θέσει ως multiple το HotelAmenties διότι θέλω να αποθηκεύεται λίστα με όλες τις υπηρεσίες που παρέχονται

TripOffer: Σε αυτό το object θα αποθηκεύσω πληροφορίες για την προσφορά ταξιδιού που προτείνει ο πράκτορας στον υπάλληλο.

Χαρακτηριστικά:

- hotelName --> Ξενοδοχείο διαμονής (τύπου String)
- amenties --> Υπηρεσίες ξενοδοχείου διαμονής (τύπου String)
- hotelCost --> Κόστος διαμονής στο ξενοδοχείο (τύπου Integer)
- distanceFromCity --> Απόσταση από το κέντρο της πόλης (τύπου Integer)
- airlineCompany --> Εταιρία αεροπλάνου (τύπου String)
- departure --> Ημερομηνία αναχώρησης(Προς προορισμό) (τύπου Datetime)
- arrival --> Ημερομηνία άφιξης (Προς προορισμό) (τύπου Datetime)
- returnDeparture --> Ημερομηνία αναχώρησης(Προς κατοικία) (τύπου Datetime)
- returnArrival --> Ημερομηνία άφιξης(Προς κατοικία) (τύπου Datetime)
- weightOfLuggage --> Καλυπτόμενο βάρος βαλίτσας (τύπου Float)
- airplaneCost --> Κόστος πτήσης (τύπου Integer)
- transitCompany --> Εταιρία υπεύθυνη για τις μεταφορές του υπαλλήλου (τύπου String)
- transitCost --> Κόστος μεταφορών (τύπου Integer)

- tripInsurance --> Boolean εάν θα διατίθεται ασφάλεια ταξιδιού (τύπου Boolean)
- finalCost --> Συνολικό κόστος πακέτου (τύπου Integer)
- isOfferAccepted --> Boolean για να σημειωθεί εάν η προσφορά είναι αποδέκτη (τύπου Boolean)
- changesNeeded --> Boolean εάν ο υπάλληλος θέλει να γίνουν αλλαγές στο πακέτο διότι δεν είναι αποδεκτοί οι όροι (τύπου Boolean)
- arrangedByTA --> Ο πράκτορας που δημιούργησε και έστειλε την προσφορά (τύπου TravelAgent)

Από αυτά τα χαρακτηριστικά έχουμε θέσει ένα ως Mandatory (Δηλαδή Υποχρεωτικό) και συγκεκριμένα είναι το 1) finalCost

Επιπλέον από αυτά τα χαρακτηριστικά έχουμε θέσει σαν λίστα μόνο τα amenities, καθώς θέλω να μου εμφανίζονται όλες οι υπηρεσίες που παρέχονται στην προσφορά του ταξιδιού.

Στο χαρακτηριστικό arrangedByTA έχουμε θέσει τον τύπο του σαν TravelAgent(Object) και το Relationship το έχουμε θέσει σαν "Composition"

ΚΕΦΑΛΑΙΟ 5 : ΘΕΜΑΤΑ ΥΛΟΠΟΙΗΣΗΣ

BUSINESS VARIABLES, PROCESS VARIABLES, TIMER EVENTS, MESSAGE EVENTS, USER-ACTOR FILTERS, GATEWAYS

GATEWAYS

- Is Request Accepted?** : Στο συγκεκριμένο gateway εξετάζεται η απόφαση του manager και συγκεκριμένα το αν η αίτηση που έστειλε ο employee για ταξίδι χορηγούμενο από την εταιρία έγινε δεκτή ή όχι. Συγκεκριμένα θέτω ως Default flow την εκδοχή του να έχει εγκριθεί η αίτηση από τον manager και για την περίπτωση που η αίτηση απορρίφθηκε στο βέλος ροής που οδηγεί στην ενημέρωση του employee πως η αίτηση του απορρίφθηκε θα ενσωματώσω το εξής condition : `travelRequestObj.isRequestAccepted==false`
 Αυτό σημαίνει πως αν ο manager επιλέξει να απορρίψει την αίτηση του employee επειδή ή τιμή της μεταβλητής isRequestAccepted θα έχει πάρει την τιμή false η ροή της διαδικασίας θα οδηγηθεί προς την ενημέρωση του employee ότι η αίτησή του απορρίφθηκε και θα τερματίζεται η διαδικασία.
- Which Message Was Sent?** : Αυτό το gateway λειτουργεί σαν “σκαλοπάτι” για την ομαλή συνέχεια της ροής. Συγκεκριμένα η χρήση του συγκεκριμένου gateway εξασφαλίζει πως μόνο μια από τις 2 εισερχόμενες ροές μπορεί να συνεχίσει προς το επόμενο βήμα της διαδικασίας.
- Which Message Was Received?** : Αντίστοιχα με το προηγούμενο gateway έτσι και σε αυτό εξασφαλίζουμε πως μόνο μια από τις 3 εισερχόμενες ροές μπορούν να εισέλθουν στο συγκεκριμένο gateway και να οδηγήσουν τη διαδικασία στο επόμενο βήμα της.
- Is Offer Accepted?** : Όπως και το πρώτο gateway έτσι και εδώ εξετάζεται το ποια ροή θα ακολουθηθεί σύμφωνα με την απάντηση που έδωσε ο employee στην προσφορά που έλαβε από τον travel agent. Στο βέλος ροής με ονομασία Yes τοποθετούμε το condition `replyToOfferReceivedByAgent.isOfferAccepted==true` . Αυτό πρακτικά σημαίνει πως στην περίπτωση που ο employee έχει απαντήσει θετικά στην πρόταση που του έγινε θα ακολουθηθεί η συγκεκριμένη ροή. Αλλιώς στην περίπτωση που η μεταβλητή isOfferAccepted δεν έχει λάβει την τιμή true θα ακολουθηθεί η default ροή και η διαδικασία θα οδηγηθεί στην αναζήτηση νέων προσφορών από τον travel agent.

- **After Acceptance 1** : Με την χρήση του συγκεκριμένου gateway επιτυγχάνουμε την παράλληλη διεξαγωγή των 2 επόμενων task Make Reservation και Arrange Trip Insurance.
- **After Acceptance 2** : Με τη χρήση του συγκεκριμένου gateway επιτυγχάνουμε το γεγονός πως η διαδικασία δεν θα προχωρήσει στο επόμενο βήμα μέχρις ότου να ολοκληρωθούν και τα 2 προηγούμενα user tasks.
- **Is Payment Complete** : Με το συγκεκριμένο gateway εξετάζουμε το ποια ροή θα ακολουθηθεί σύμφωνα με το αν ολοκληρώθηκε ή όχι η πληρωμή από τον employee. Στο βέλος ροής με ονομασία Yes τοποθετούμε το condition `expensesCoveredReceivedByAgent.areExpensesCovered==true` . Αυτό σημαίνει πως στην περίπτωση που ο employee ολοκληρώσει την πληρωμή θα ακολουθηθεί η συγκεκριμένη ροή και η διαδικασία θα οδηγηθεί στο επόμενο βήμα που είναι η δημιουργία των trip details. Σε αντίθετη περίπτωση που ο employee δεν πληρώσει το απαιτούμενο ποσό τότε θα ακολουθηθεί η default ροή και η διαδικασία θα οδηγηθεί σε ακύρωση της κράτησης και τερματισμό.

Actor Filters

Σε ένα Human task, τα actor filters είναι υπεύθυνα για το φιλτράρισμα των χρηστών που μπορούν να εκτελέσουν το συγκεκριμένο task.

Έτσι λοιπόν στο **Employee Lane** έχουμε τοποθετήσει ένα actor filter ότι ο employee είναι ο initiator. Αυτό πρακτικά σημαίνει πως όλο το process μπορεί να το ξεκινήσει μόνο ο employee και όταν η διαδικασία βρίσκεται σε εξέλιξη δεν μπορεί κάποιος άλλος employee ,εκτός από αυτόν που εκκίνησε τη διαδικασία, να δει και/ή να επηρεάσει την εξέλιξή της.

Στο **Administration Lane** δεν ορίσαμε κάποιο actor filter στο lane όμως σε όλα τα user task πέραν του πρώτου έχουμε ορίσει φίλτρο task performer. Με αυτόν τον τρόπο επιτυγχάνουμε το εξής, το πρώτο user task που υπάρχει στο lane του administration μπορεί να το εκτελέσει οποιοσδήποτε από τους διαθέσιμους managers της εταιρίας. Όμως όλα τα υπόλοιπα human task του συγκεκριμένου lane δεν μπορεί να τα εκτελέσει οποιοσδήποτε manager παρά μόνο ο συγκεκριμένος που εκτέλεσε και το πρώτο.

Αντίστοιχα στο **Finance Department Lane** έχουμε ορίσει το task performer filter στο Send Payment Human task έτσι ώστε να εξασφαλίσουμε

πως ο accountant που θα ολοκληρώσει την επιστροφή χρημάτων στον employee θα είναι ο ίδιος με αυτόν που θα λάβει την εντολή από τον manager και θα συμπληρώσει τη φόρμα με τα στοιχεία του.

Με την ίδια λογική λειτουργήσαμε και στο Lane του **Travel Agent** καθώς έχουμε βάλει το task performer φίλτρο σε όλα τα human tasks πέραν του πρώτου έτσι ώστε να εξασφαλίσουμε πως με τη συγκεκριμένη διαδικασία θα ασχολείται μόνο ένας travel agent καθ' όλη τη διάρκειά της. Μοναδική εξαίρεση αποτελεί το Create Reminder human task το οποίο με τη χρήση start message event και end message event του έχουμε δώσει την ιδιότητα να είναι ξεχωριστό process οπότε δεν μπορούμε να του ορίσουμε φίλτρο που να το αναγκάζει να εκτελεστεί από τον agent διαφορετικού process.

Timer Events

Στο διάγραμμά μας έχουμε χρησιμοποιήσει 2 φορές timer event το κάθε ένα με διαφορετικό σκοπό.

Το πρώτο είναι το 5-Day Duration το οποίο είναι ένα boundary timer event πάνω στο human task review trip offer. Με τον συγκεκριμένο τρόπο χρήσης του timer event βάζουμε ένα exception στο human task έτσι ώστε αν αυτό δεν ολοκληρωθεί (να στείλει δηλαδή ο employee απάντηση στον travel agent για την προσφορά που έλαβε) στο διάστημα των 5 ημερών (το οποίο είναι και το αντίστοιχο timer condition που έχουμε θέσει) τότε ενεργοποιείται το exception και δημιουργείται μία εναλλακτική ροή η οποία θα οδηγήσει στον travel agent ο οποίος θα δημιουργήσει και θα στείλει στον employee ένα ενημερωτικό μήνυμα που θα του υπενθυμίζει την ύπαρξη της προσφοράς και θα του ζητά να απαντήσει σε αυτήν.

Το δεύτερο timer event που χρησιμοποιούμε βρίσκεται προς το τέλος του diagram και είναι intermediate timer event και σε αντίθεση με το προηγούμενο αυτό δεν λειτουργεί ως exception αλλά κάνει pause τη ροή της διαδικασίας μέχρι να τελειώσει η αντίστροφη μέτρηση που του έχουμε ορίσει. Συγκεκριμένα το human task ακριβώς πριν από το timer είναι το Complete Trip του οποίου η χρησιμότητα είναι να ορίσει ο employee πότε ακριβώς επέστρεψε από το ταξίδι του και έτσι όταν γίνει αυτό το timer θα ξεκινήσει να μετράει αντίστροφα για 1 ημέρα (timer condition) έτσι ώστε να υπάρχει το απαραίτητο κενό μιας ημέρας από την επιστροφή του employee μέχρι την αποστολή των εξόδων του προς τη διοίκηση της εταιρίας.

Message Events

Διαδικασία μεταφοράς δεδομένων μέσω μηνυμάτων (message events):

Ακολουθούμε τον ίδιο τρόπο μεταφοράς δεδομένων ανάμεσα στα δύο pools, στο Company και στο Travel Agency. Η ίδια διαδικασία θα γίνει σε # παραδείγματα.

Το πρώτο παράδειγμα αφορά την μεταφορά της αίτησης για ταξίδι του υπαλλήλου στον Travel Agent

Ξεκινάμε με ένα throw message event που ονομάζεται Send Request Offer και καταλήγει στο start message event με όνομα Receive Request Offer και μέσα μεταφέρεται το message requestoffer. Το requestoffer περιέχει το Persistenceld του TravelRequestObj(συνεπώς είναι και τύπου long), το οποίο έχει συμπληρωθεί από τον υπάλληλο και ελεγχθεί από τον υπεύθυνο. Το requestOffer() script που κάνει αυτή τη διαδικασία είναι :
travelRequestObj.getPersistenceld(). Όλο το item ονομάζεται transitdata.

Όταν το μήνυμα ληφθεί στο Receive Request Offer, δημιουργούμε στο message content μία μεταβλητή long την receivedRequestPERID να παίρνει τη τιμή του περιεχομένου του transitdata.

Στην συνέχεια, θα χρειαστούμε ένα Service Task(Build Business Var From Persistence Id Received) για να κάνουμε ένα data object και πιο συγκεκριμένα το TravelRequest να έχει τις συμπληρωμένες τιμές της φόρμας που του στείλαμε από το Company. Η σύνδεση θα γίνει στο operations φτιάχνοντας το travelRequestReceivedByAgent TravelRequest Object, το οποίο παίρνει τα συμπληρωμένα στοιχεία που υπάρχουν περασμένα μέσα από το Persistenceld που είναι μοναδικό. Το script που το κάνει αυτό είναι το travelRequestByPERID(). Το περιεχόμενο του είναι:

```
return travelRequestDAO.findByPersistenceId(receivedRequestPERID);
```

Τέλος χρησιμοποιούμε το αντικείμενο TravelRequest προσθέτοντας από τα δεδομένα το travelRequestReceivedByAgent.

- Για την μεταφορά της αρχικής προσφοράς ταξιδιού χρησιμοποιήθηκαν ένα throw message event και ένα catch message event, με ονόματα Send Initial Offer και Receive Initial Offer αντίστοιχα.
Το μήνυμα που μεταφέραν ήταν το InitialOffer με περιεχόμενο το item initialOfferedPerId, με την τιμή του
tripOfferPerIdCreated():
tripOffer.getPersistenceId()

Μετά δημιουργούμε στο message content μία μεταβλητή `tripOfferReceivedPERID` που δέχεται ως τιμή το μήνυμα.

Στο Service Task συνδέουμε το Data Object τύπου `tripOffer` με ονομασία `tripOfferReceiveByEmployee` με τιμή

```
tripOfferCon():
```

```
return tripOfferDAO.findByPersistenceId(tripOfferReceivedPERID);
```

- Για την μεταφορά των επόμενων προσφορών ταξιδιού χρησιμοποιήθηκαν ένα throw message event και ένα catch message event, με ονόματα Send Alternative Offer και Receive Alternative Offer αντίστοιχα.

Το μήνυμα που μεταφέραν ήταν το AlternativeOffer με περιεχόμενο το item `alternativeOfferPerId`, με την τιμή του

```
altOfferPerId():tripOffer.getPersistenceId()
```

Μετά δημιουργούμε στο message content μία μεταβλητή

`alternativeOfferReceivedPERID` που δέχεται ως τιμή το μήνυμα.

Στο Service Task με όνομα: Copy of Build Business Var From Persistence Id Received συνδέουμε το Data Object τύπου `tripOffer` με ονομασία `tripOfferReceiveByEmployee` με τιμή

```
alttripOfferCon():
```

```
return tripOfferDAO.findByPersistenceId(alternativeOfferReceivedPERID);
```

- Για την μεταφορά της υπενθύμισης χρησιμοποιήθηκαν ένα end message event και ένα catch message event, με ονόματα Send Reminder και Receive Reminder αντίστοιχα.

Το μήνυμα που μεταφέραν ήταν το reminder με περιεχόμενο το item `reminderPERID`, με την τιμή του

```
remPERID():reminderCreatedByAgent.getPersistenceId()
```

Μετά δημιουργούμε στο message content μία μεταβλητή `reminderPERID` που δέχεται ως τιμή το μήνυμα.

Στο Service Task με όνομα: Copy of Build Business Var From Persistence Id Received συνδέουμε το Data Object τύπου `tripOffer` με ονομασία `reminderReceivedByEmployee` με τιμή

```
reminderCon():
```

```
return reminderDAO.findByPersistenceId(reminderPERID);
```

- Για την μεταφορά της απάντησης του υπαλλήλου στην προσφορά του travel agent χρησιμοποιήθηκαν ένα throw message event και ένα catch message event, με ονόματα Send Review Reply και Receive Review Reply αντίστοιχα.

Το μήνυμα που μεταφέραν ήταν το Offer Reply με περιεχόμενο το item `replyToOffer`, με την τιμή του

```
receivedFromEmpOffer():tripOfferReceiveByEmployee.getPersistenceId()
```

Μετά δημιουργούμε στο message content μία μεταβλητή `replytoOfferPERID` που δέχεται ως τιμή το μήνυμα.

Στο Service Task με όνομα: Build Business Var From Persistence Id Received 2 συνδέουμε το Data Object τύπου `tripOffer` με ονομασία `replyToOfferReceivedByAgent` με τιμή

```
tripOfferToTACon():
return tripOfferDAO.findByPersistenceId(replytoOfferPERID);
```

- Για την μεταφορά του κόστους χρησιμοποιήθηκαν ένα throw και ένα catch message event, με ονόματα Send Bill και Receive Bill αντίστοιχα. Το μήνυμα που μεταφέραν ήταν το Send Bill με περιεχόμενο το item `billSentByAgent`, με την τιμή του

```
createexpe():expensesCreatedByAgent.getPersistenceId()
```

Μετά δημιουργούμε στο message content μία μεταβλητή `expePERID` που δέχεται ως τιμή το μήνυμα.

Στο Service Task με όνομα: Build Business Var From Expenses Persistence Id Received συνδέουμε το Data Object τύπου `expenses` με ονομασία `replyToOfferReceivedByAgent` με τιμή

```
expeCon():
return expensesDAO.findByPersistenceId(expePERID);
```

- Για την μεταφορά της πληρωμής χρησιμοποιήθηκαν ένα throw και ένα catch message event, με ονόματα Send Payment και Receive Payment αντίστοιχα.

Το μήνυμα που μεταφέραν ήταν το Payment με περιεχόμενο το item `PaymentSentByEmployee`, με την τιμή του

```
sendpayment():expensesCreatedByAgent.getPersistenceId()
```

Μετά δημιουργούμε στο message content μία μεταβλητή `paymentReceived` που δέχεται ως τιμή το μήνυμα.

Στο Service Task με όνομα Completed Payment Check συνδέουμε το Data Object τύπου `expenses` με ονομασία `replyToOfferReceivedByAgent` με τιμή

```
paymentCon():
return expensesDAO.findByPersistenceId(paymentReceived);
```

- Για την μεταφορά των λεπτομερειών του ταξιδιού χρησιμοποιήθηκαν ένα throw και ένα catch message event, με ονόματα Send Trip Details και Receive Trip Details αντίστοιχα.

Το μήνυμα που μεταφέραν ήταν το Trip Details με περιεχόμενο το item `tripDetailsSentByAgent`, με την τιμή του

```
tripDetailsPERID():tripDetails.getPersistenceId()
```

Μετά δημιουργούμε στο message content μία μεταβλητή `tripDetails` που δέχεται ως τιμή το μήνυμα.

Στο Service Task με όνομα Build Business Var From Persistence Id Received 5 συνδέουμε το Data Object τύπου tripDetails με ονομασία receivedTripDetailsByEmployee με τιμή

```
tripDetailsCon():  
return tripDetailsDAO.findByPersistenceId(tripDetails)
```

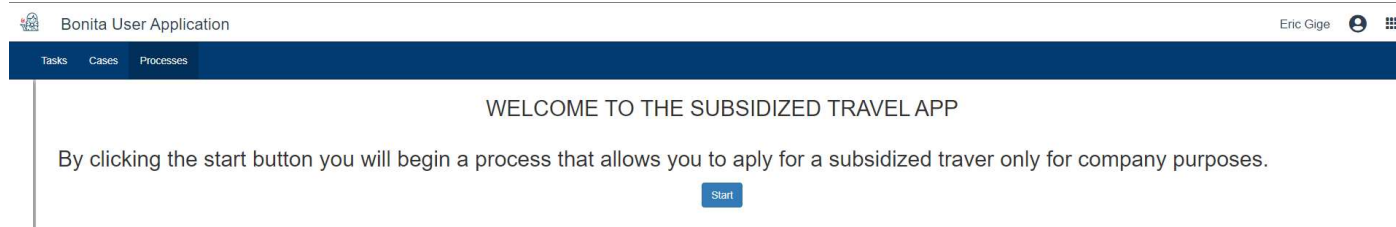
Το μόνο μήνυμα που δεν περιέχει item μέσα και γίνεται με πιο απλή διαδικασία είναι το παρακάτω:

Για την ειδοποίηση ότι πέρασαν 5 μέρες μετά το 5-Day Duration timer, χρησιμοποιούμε ένα throw και start message event, με ονομασίες 5 days duration has passed και 5days notification received by travel agent αντίστοιχα. Το μήνυμα που μεταφέρεται είναι το StartNotification το οποίο δεν περιέχει item μέσα.

ΚΕΦΑΛΑΙΟ 6 : ΕΓΧΕΙΡΙΔΙΟ ΧΡΗΣΤΗ

ΕΓΧΕΙΡΙΔΙΟ ΧΡΗΣΗΣ ΜΕ ΠΕΡΙΓΡΑΦΗ ΟΛΩΝ ΤΩΝ ΟΘΟΝΩΝ ΤΟΥ ΣΥΣΤΗΜΑΤΟΣ.

Ξεκινώντας την εφαρμογή εμφανίζεται η σελίδα εκκίνησης η οποία αναγράφει τον τίτλο του process και δίνει την δυνατότητα στον χρήστη να επιλέξει πότε εκείνος θέλει να την ξεκινήσει



Αφού ο χρήστης είναι συνδεδεμένος στην εφαρμογή και ξεκινήσει η διαδικασία, ο employee αρχίζει να συμπληρώνει τα στοιχεία της φόρμας αίτησης για το ταξίδι. Αφού συμπληρώσει όλα τα απαιτούμενα στοιχεία του form και τα comments, πατάει "submit" ώστε να γίνει η υποβολή της φόρμας και πλέον περιμένει για την απάντηση από τον manager του.

Bonita User Application Eric Gige

Tasks Cases Processes

Form Comments

Surname
Gige

Job Title
Marketing Coordinator

Department
Marketing

Email *
ericgige@gmail.com

Phone Number *
6953264156

Home Phone Number
2105966432

Fax
2105966432

Address Street
25is Martiou

Address Number
5

Close

Bonita User Application Eric Gige

Tasks Cases Processes

Form Comments

Travel Request Form

Applier Details Travel Details

Travel Details

Leaves On *
02/25/2023 Today 3:20:00 PM Now

Returns On *
03/04/2023 Today 3:20:00 PM Now

Country *
Spain

City *
Madrid

Travel Reason *
Marketing Seminar

Submit

Close

Όταν πλέον ο manager συνδεθεί στον λογαριασμό του θα του εμφανιστεί η αντίστοιχη αίτηση και θα πρέπει αρχικά να συμπληρώσει τα προσωπικά του στοιχεία και στη συνέχεια να επεξεργαστεί την αίτηση του employee και να αποφασίσει αν θα αποδεχτεί ή όχι την αίτηση. Επιπλέον επιβεβαιώνει ότι είναι ο υπεύθυνος για τον αιτούμενο εργαζόμενο και πατάει submit.

Bonita User Application John Coe

Tasks Cases Processes

Form Comments

Verify Travel Request

Travel Request Details Manager Details

Travel Details

Leaves On *
02/25/2023 Today 3:20:00 PM Now

Returns On *
03/04/2023 Today 3:20:00 PM Now

Country *
Spain

City *
Madrid

Travel Reason *
Marketing Seminar

Applier Details

Employee Id *
employee1

Name
Eric

Surname
Gige

Job Title
Marketing Coordinator

Department
Marketing

Email *
ericgige@gmail.com

Phone Number *
6953284156

Close

Bonita User Application John Coe

Tasks Cases Processes

Filters

Process All Case Case ID

Search...

Task list

Task name	Process name	Due date
Review Travel Request	Company	-

1 - 1 / 1

Form Comments

Verify Travel Request

Travel Request Details Manager Details

Manager Details

Manager Surname
Coe

Manager Name
John

Manager Id *
manager1

Manager Department
Marketing

Manager Job Title
Marketing Chief

Manager Email *
johncoe@gmail.com

Manager Phone Number *
6934568721

Manager Home Number
2114586379

Manager Address Street
Thivon

Bonita User Application John Coe

Tasks Cases Processes

Filters

Process All Case Case ID

Search...

Task list

Task name	Process name	Due date
Review Travel Request	Company	-

1 - 1 / 1

Form Comments

Coe

Manager Name
John

Manager Id *
manager1

Manager Department
Marketing

Manager Job Title
Marketing Chief

Manager Email *
johncoe@gmail.com

Manager Phone Number *
6934568721

Manager Home Number
2114586379

Manager Address Street
Thivon

Manager Address Number
123

☒ I am the authorized manager that supervises this employee

☒ Trip Request is Accepted

Submit

Στην περίπτωση που ο υπεύθυνος δεν θέλει να αποδεχτεί την αίτηση του υπαλλήλου και δεν κλικάρει το κουτί με όνομα «Trip Request is Accepted», στο επόμενο βήμα εμφανίζεται από το προφίλ του υπαλλήλου η ανακοίνωση της άρνησης της αιτήσεως του με όνομα task “Notify Employee For Request Denial”. Εκεί του εμφανίζονται στα Tab 1-3 τα στοιχεία που συμπλήρωσε ο ίδιος προηγουμένως και τα στοιχεία του υπεύθυνου που ακύρωσε την αίτηση και τέλος στο Tab 4 εμφανίζονται τα παρακάτω. Ο χρήστης δεν μπορεί να τσεκάρει το τετράγωνο του «Trip Request is Accepted» όπως και όλα τα στοιχεία αυτής της φόρμας καθώς είναι μόνο για ανακοίνωση αποτελεσμάτων. Το μόνο που μπορεί να κάνει ο χρήστης είναι να τα διαβάσει και στο τέλος να πατήσει Continue. Πατώντας το Continue, δεν εμφανίζεται τίποτα άλλο σε κανέναν εργαζόμενο και η διαδικασία φτάνει στο τέλος της.

Έστω ότι συνεχίζουμε με την περίπτωση ότι ο manager αποδέχτηκε την αίτηση. Έτσι λοιπόν ο employee όταν ξανασυνδεθεί στον λογαριασμό του θα λάβει το σχετικό μήνυμα ότι η αίτησή του έγινε δεκτή και θα πρέπει πλέον να αποστείλει την αίτησή του στο πρακτορείο ταξιδιών (Form Offer Request To Agent), πατώντας μόνο το Send κουμπί.

Έτσι λοιπόν στο πρακτορείο πλέον έχει σταλθεί η προσφορά και όταν κάποιος από τους πράκτορες συνδεθεί στον λογαριασμό του θα δει την συγκεκριμένη αίτηση και θα πρέπει να πατήσει την ένδειξη take για να πάρει την ευθύνη του task και να συνεχίσει.

The screenshot shows the Bonita User Application interface. On the left, there is a sidebar with 'To do', 'My tasks', and 'Done tasks'. The main area is divided into 'Filters' and 'Task list'. The 'Task list' shows a task named 'Record Request' under the process 'Travel Agency'. The 'Take' button is highlighted. On the right, there is a form titled 'Offer Request From Employee' with fields for Employee ID, Name, Surname, Email, Phone Number, and Home Phone Number. A warning message at the top of the form states: 'To fill out the form, you need to take this task. This means you will be the only one able to do it. To make it available to the team again, release it.' Below the form, there are tabs for 'Employee Details' and 'Travel Details'.

Στη συνέχεια μέσω του κουμπιού submit κάνει record την αίτηση και προχωράει στο επόμενο βήμα της διαδικασίας.

The screenshot shows the Bonita User Application interface. On the left, there is a sidebar with 'To do', 'My tasks', and 'Done tasks'. The main area is divided into 'Filters' and 'Task list'. The 'Task list' shows a task named 'Record Request' under the process 'Travel Agency'. The 'Submit' button is highlighted. On the right, there is a form titled 'Offer Request From Employee' with fields for Employee ID, Name, Surname, Email, Phone Number, and Home Phone Number. Below the form, there are tabs for 'Employee Details' and 'Travel Details'. The 'Travel Details' tab is active, showing fields for 'Leaves On' (02/25/2023), 'Returns On' (03/04/2023), 'Country' (Spain), and 'City' (Madrid). A 'Submit' button is at the bottom of the form.

Αφού γίνει το record τότε ο agent πρέπει να δημιουργήσει ένα πακέτο προσφοράς για τον πελάτη, αφού πρώτα συμπληρώσει τα προσωπικά του στοιχεία.

Bonita User Application Eva Old

Tasks Cases Processes

Filters

Process: All Case: Case ID

Search...

Task list

Task name	Process name	Due date
Search for Initial Offer	Travel Agency	-

1 - 1/1

Form Comments

Trip Offer

Travel Agent Details Offer Details

Authorized Travel Agent

Travel Agent Name *
Eva

Travel Agent Surname *
Old

Travel Agent Email *
evaxold@gmail.com

Travel Agent Phone Number *
6958741258

Travel Agent Fax
2114735685

Travel Agent Address Street
Kyprou

Travel Agent Address Number
8

Bonita User Application Eva Old

Tasks Cases Processes

Filters

Process: All Case: Case ID

Search...

Task list

Task name	Process name	Due date
Search for Initial Offer	Travel Agency	-

1 - 1/1

Form Comments

Trip Offer

Travel Agent Details Offer Details

Offer

Hotel Name
Riu Plaza España

Hotel Cost
179

Distance From City
200

Airline Company
Wizz Air UK

Departure
02/25/2023 Today 3:40:00 PM Now

Arrival
02/25/2023 Today 7:40:00 PM Now

Return Departure
03/04/2023 Today 3:40:00 PM Now

Return Arrival
02/25/2023 Today 7:40:00 PM Now

Weight Of Luggage
18

Airplane Cost
366

Transit Company
Its In Transit

Transit Cost
50

Final Cost *
595

Submit

Bonita User Application Eva Old

Tasks Cases Processes

Filters

Process: All Case: Case ID

Search...

Task list

Task name	Process name	Due date
Search for Initial Offer	Travel Agency	-

1 - 1/1

Form Comments

Trip Offer

Travel Agent Details Offer Details

Wizz Air UK

Departure
02/25/2023 Today 3:40:00 PM Now

Arrival
02/25/2023 Today 7:40:00 PM Now

Return Departure
03/04/2023 Today 3:40:00 PM Now

Return Arrival
02/25/2023 Today 7:40:00 PM Now

Weight Of Luggage
18

Airplane Cost
366

Transit Company
Its In Transit

Transit Cost
50

Final Cost *
595

Submit

Αφού έχει συμπληρώσει όλα τα απαραίτητα στοιχεία της προσφοράς μέσω του submit στέλνει την συγκεκριμένη προσφορά στον employee της εταιρίας και περιμένει πλέον την απάντησή του.

Στην ενδεχόμενη περίπτωση που ο employee καθυστερήσει πάνω από 5 ημέρες να απαντήσει στην συγκεκριμένη πρόταση τότε ο travel agent δημιουργεί ένα ειδοποιητικό μήνυμα προς τον employee και του το στέλνει πατώντας το submit. (σ.σ. για να μην χρειάζεται πραγματική αναμονή 5 ημερών κατά τη διάρκεια της διεξαγωγής του συγκεκριμένου παραδείγματος μειώσαμε το boundary timer στο task Review Offer στα 10 δευτερόλεπτα και εν συνεχεία για τους σκοπούς της άσκησης το αποκαταστήσαμε στις 5 ημέρες.)

The screenshot shows the Bonita User Application interface. On the left, there's a sidebar with 'To do', 'My tasks', and 'Done tasks'. The main area is divided into 'Filters' and 'Task list'. The 'Task list' shows a task named 'Create Reminder' for the 'Travel Agency' process. On the right, the 'Form' tab is active, displaying a 'Friendly Reminder' form. The form text reads: 'On behalf of the partner travel agency we would like to know if you are still interested in the offer we sent you as the 5 day time limit has passed without receiving a response.' Below the text is a checkbox labeled 'Is Client Still interested?' and a 'Send Reminder' button.

Στον employee πλέον έχει εξαφανιστεί η προσφορά και φανερώνεται το μήνυμα στο οποίο πρέπει να κλικάρει την ένδειξη και να πατήσει το submit για να προχωρήσει η διαδικασία.

The screenshot shows the Bonita User Application interface. On the left, the sidebar is the same. The 'Task list' now shows a task named 'Review Reminder' for the 'Company' process. On the right, the 'Form' tab is active, displaying a 'Friendly Reminder' form. The form text reads: 'On behalf of the partner travel agency we would like to know if you are still interested in the offer we sent you as the 5 day time limit has passed without receiving a response.' Below the text is a checkbox labeled 'I confirm I was informed and I will send my offer reply in the next 5 days' and a 'Submit' button.

Αφού ο employee δήλωσε πλέον ότι συνεχίζει να ενδιαφέρεται για το συγκεκριμένο ταξίδι του εμφανίζεται ξανά η προσφορά την οποία αξιολογεί και διαλέγει εάν θα την αποδεχτεί ή εάν θα ζητήσει αλλαγές.

Bonita User Application Eric Gige

Tasks Cases Processes

Filters

Process: All

Case: Case ID

Search...

Task list

Take Release

Task name	Process name	Due date
Review Trip Offer	Company	-

Form submitted. The next task in the list is now selected.

Travel Offer From Travel Agency

Travel Agent Details Offer Details

Travel Agent Details

Name *
Eva

Surname *
Old

Email *
evaid@gmail.com

Phone Number *
2114735685

Fax
2114735685

Bonita User Application Eric Gige

Tasks Cases Processes

Filters

Process: All

Case: Case ID

Search...

Task list

Take Release

Task name	Process name	Due date
Review Trip Offer	Company	-

Form Comments

03/04/2023 Today 3:40:00 PM Now

Return Arrival
02/25/2023 Today 7:40:00 PM Now

Weight Of Luggage
18

Airplane Cost
366

Transit Company
Its In Transit

Transit Cost
50

Trip Insurance

Final Cost *
595

Please check only one from the two boxes, the one representing your view of this offer.

☒ I Accept This Offer

☐ I Request Changes In This Offer

Submit









Στην περίπτωση που ο employee δεν μείνει ικανοποιημένος από την προσφορά του agent τότε θα τσεκάρει το "I Request Changes In This Offer" τότε στον Travel Agent εμφανίζεται το "Search for Alternative Offers" όπου είναι η ίδια ακριβώς φόρμα που εμφάνιζε στο «Search for Initial Offer», την ξανασυμπληρώνει και αυτή η διαδικασία μπορεί να επαναληφθεί επ' αορίστον μέχρι να αποδεχτεί την προσφορά ο υπάλληλος.

Please check only one from the two boxes, the one representing your view of this offer.

☐ I Accept This Offer

☒ I Request Changes In This Offer

Submit

Form	Comments
Departure	
02/25/2023	Today  4:05:00 PM Now 
Arrival	
02/25/2023	Today  8:05:00 PM Now 
Return Departure	
03/04/2023	Today  7:10:00 PM Now 
Return Arrival	
03/04/2023	Today  11:00:00 PM Now 
Weight Of Luggage	
<input type="text" value="18"/>	
Airplane Cost	
<input type="text" value="200"/>	
Transit Company	
<input type="text" value="Madrid Fast Travels"/>	
Transit Cost	
<input type="text" value="78"/>	
<input checked="" type="checkbox"/> Trip Insurance	
Final Cost ^	
<input type="text" value="378"/>	
<input type="button" value="Submit"/>	

Έστω ότι συνεχίζουμε με την περίπτωση πως ο employee αποδέχτηκε την αρχική προσφορά του travel agent έτσι όταν πλέον ο travel agent ξανά συνδεθεί στον λογαριασμό του θα πρέπει αυτός πλέον να συνεχίσει τη διαδικασία. Αρχικά θα πρέπει αν επιβεβαιωθεί την χρήση check box ότι είναι η ασφάλεια του ταξιδιού είναι έτοιμη και θα πατήσει submit για να αποθηκευτεί.

The screenshot shows the Bonita User Application interface. On the left, there's a sidebar with 'Tasks', 'Cases', and 'Processes' tabs. Below them are 'To do', 'My tasks', and 'Done tasks' sections. The main area is divided into 'Filters' and 'Task list'. The 'Filters' section has a 'Process' dropdown set to 'All', a 'Case' dropdown, and a 'Case ID' input field. Below this is a search bar. The 'Task list' section shows a table with columns 'Task name', 'Process name', and 'Due date'. The table contains two tasks: 'Arrange Trip Insurance' and 'Make Reservation', both assigned to 'Travel Agency'. The 'Arrange Trip Insurance' task is highlighted. On the right, the 'Form' tab is active, showing the 'Trip Insurance' form. It has a 'Form' tab and a 'Comments' tab. The form content includes a checkbox labeled 'Trip Insurance is Set' and a 'Submit' button.

Ενώ παράλληλα θα πρέπει να επικυρωθεί το reservation οπότε ο πράκτορας θα επιλέξει το check box και έπειτα θα πατήσει submit για να αποθηκευτεί.

The screenshot shows the Bonita User Application interface. On the left, there's a sidebar with 'Tasks', 'Cases', and 'Processes' tabs. Below them are 'To do', 'My tasks', and 'Done tasks' sections. The main area is divided into 'Filters' and 'Task list'. The 'Filters' section has a 'Process' dropdown set to 'All', a 'Case' dropdown, and a 'Case ID' input field. Below this is a search bar. The 'Task list' section shows a table with columns 'Task name', 'Process name', and 'Due date'. The table contains one task: 'Make Reservation', assigned to 'Travel Agency'. The 'Make Reservation' task is highlighted. On the right, the 'Form' tab is active, showing the 'Complete Reservation' form. It has a 'Form' tab and a 'Comments' tab. The form content includes a checkbox labeled 'Reservation is completed' and a 'Submit' button.

Αφού ολοκληρωθούν αυτά τα 2 task τότε ο travel agent καλείται να δημιουργήσει τον τελικό λογαριασμό του ταξιδιού και μέσω του submit να τον στείλει στον πελάτη.

Bonita User Application Eva Old

Tasks Cases Processes

Filters

To do 1

My tasks 1

Done tasks

Process All Case Case ID

Search...

Task list

Take Release 1 - 1 / 1

Task name	Process name	Due date
Create Bill	Travel Agency	-

1 - 1 / 1

Create Trip Bill

Trip Expenses

Hotel Cost 179

Transit Cost 50

Airplane Cost 366

Final Cost * 595

Submit

Ο πελάτης πλέον έχει λάβει τον τελικό λογαριασμό του ταξιδιού και πρέπει να αποφασίσει αν θα πληρώσει το τελικό ποσό ή όχι.

Bonita User Application Eric Gige

Tasks Cases Processes

Filters

To do 1

My tasks 1

Done tasks

Process All Case Case ID

Search...

Task list

Take Release 1 - 1 / 1

Task name	Process name	Due date
Pay Bill	Company	-

1 - 1 / 1

Trip Bill

Trip Expenses

Hotel Cost 179

Transit Cost 50

Airplane Cost 366

Final Cost * 595

Leave "Are Expenses Covered" option unchecked if you don't want to pay the bill. If you do so the trip will be canceled.

☒ Are Expenses Covered

Submit

Στην περίπτωση που δεν τσεκαριστεί το κουτί «Are Expenses Covered» ακυρώνεται όλη η διαδικασία καθώς ο πελάτης αρνείται να πληρώσει.

Leave "Are Expenses Covered" option unchecked if you don't want to pay the bill. If you do so the trip will be canceled.

☐ Are Expenses Covered

Submit

Έστω ότι συνεχίζουμε με την επιλογή πως ο employee έχει τσεκάρει το “Are Expenses Covered” και ολοκληρώνει την πληρωμή. Πλέον ο travel agent αναλαμβάνει τη συμπλήρωση φόρμας με όλες τις λεπτομέρειες του ταξιδιού και αφού συμπληρώσει όλες τις λεπτομέρειες πατάει το κουμπί submit και τις στέλνει στον employee.

Bonita User Application Eva Old

Tasks Cases Processes

Filters

Process **All** Case Case ID

Search...

Task list

Take Release 1 - 1 / 1

Task name	Process name	Due date
Create Trip Details	Travel Agency	-

1 - 1 / 1

Form Comments

Complete The Trip Details

Trip Details

Departure Details Return Details Transit-Hotel Details Trip Insurance

Airline Selected
Wizz Air UK

Flight ID *
9518

Electronic Ticket Number
13456

From Airport
El. Venizelos

To Airport
Adolfo Suárez Madrid-Barajas Airport

Departure Gate
6

Departure Seat
13

Departure Date-Time
02/25/2023 Today 3:40:00 PM Now

Arrival Date-Time
02/25/2023 Today 7:40:00 PM Now

Bonita User Application Eva Old

Tasks Cases Processes

Filters

Process **All** Case Case ID

Search...

Task list

Take Release 1 - 1 / 1

Task name	Process name	Due date
Create Trip Details	Travel Agency	-

1 - 1 / 1

Form Comments

Complete The Trip Details

Trip Details

Departure Details Return Details Transit-Hotel Details Trip Insurance

Return Airline Selected
Wizz Air UK

Return Flight ID *
9523

Return From Airport
Adolfo Suárez Madrid-Barajas Airport

Return To Airport
El. Venizelos

Return Gate
26

Return Seat
38

Return Departure Date-Time
03/04/2023 Today 3:40:00 PM Now

Return Arrival Date-Time
03/04/2023 Today 7:40:00 PM Now

Bonita User Application Eva Old

Tasks Cases Processes

<

To do 1

My tasks 1

Done tasks

Filters

Process **All** Case Case ID

Search...

In task name column

Task list

Take Release 1 - 1 / 1

<input type="checkbox"/>	Task name ^	Process name	Due date
<input type="checkbox"/>	Create Trip Details	Travel Agency	-

1 - 1 / 1

>

Form Comments

Complete The Trip Details

Trip Details

Departure Details Return Details Transit-Hotel Details Trip Insurance

Transit Spot Pick Up *

Gate Exit

Return Transit Spot Pick Up *

Gate Exit

Hotel Name *

Riu Plaza España

Hotel Address Street

Calle Gran Vía

Hotel Address Number

84

Hotel Type Of Room

Deluxe King Room

Bonita User Application Eva Old

Tasks Cases Processes

<

To do 1

My tasks 1

Done tasks

Filters

Process **All** Case Case ID

Search...

In task name column

Task list

Take Release 1 - 1 / 1

<input type="checkbox"/>	Task name ^	Process name	Due date
<input type="checkbox"/>	Create Trip Details	Travel Agency	-

1 - 1 / 1

>

Form Comments

Complete The Trip Details

Trip Details

Departure Details Return Details Transit-Hotel Details Trip Insurance

☒ Trip Insurance

☒ Agreement To Terms And Conditions

Submit

Συνεπώς ο employee έχει πλέον όλες τις πληροφορίες του ταξιδιού του και πρέπει να πατήσει το κουμπί submit για να προχωρήσει η διαδικασία. (Δεν παραδίδονται screenshot για το review των λεπτομερειών από τον employee καθώς είναι ακριβώς τα ίδια στοιχεία με τις παραπάνω φωτογραφίες απλώς σε read-only μορφή.

Ο employee έχει ολοκληρώσει την διαδικασία που απαιτείται για πριν το ταξίδι και αφού ολοκληρωθεί το ταξίδι του και επιστρέψει τότε θα τσεκάρει την ένδειξη “Is Trip Completed” και θα πατήσει το submit. Μετά το submit υπάρχει μια αναμονή μιας ημέρας μέχρι το επόμενο task όμως για λόγους συντομίας κατά τη διάρκεια του συγκεκριμένου παραδείγματος μειώσαμε το timer event στα 5 δευτερόλεπτα και εν συνεχεία για τους σκοπούς της άσκησης το επαναφέραμε στην 1 ημέρα.

The screenshot shows the Bonita User Application interface. The top navigation bar includes 'Tasks', 'Cases', and 'Processes'. The left sidebar shows 'To do', 'My tasks', and 'Done tasks'. The main area is titled 'Complete Trip' and contains a 'Form' tab. The form has a checkbox labeled 'Is trip Completed?' and a 'Submit' button. The left sidebar also shows a 'Task list' with a table of tasks.

Task name	Process name	Due date
Complete Trip	Company	-

Αφού λοιπόν ο employee ολοκληρώσει αυτό το βήμα και περάσει και το χρονικό περιθώριο της 1 ημέρας, τότε πρέπει να στείλει τα έξοδα του ταξιδιού στον υπεύθυνό του μέσω του κουμπιού send.

The screenshot shows the Bonita User Application interface. The top navigation bar includes 'Tasks', 'Cases', and 'Processes'. The left sidebar shows 'To do', 'My tasks', and 'Done tasks'. The main area is titled 'Send Expenses To Manager' and contains a 'Form' tab. The form has several input fields for expenses and a 'Submit' button. The left sidebar also shows a 'Task list' with a table of tasks.

Task name	Process name	Due date
Create Expenses Form	Company	-

Όταν ο manager συνδεθεί στον λογαριασμό του θα δει τον λογαριασμό που θα του έχει στείλει ο employee και στο κενό Covered By Administration Cost θα πρέπει να συμπληρώσει το ποσό που θα επιστραφεί στον employee και εν συνεχεία μέσω του submit θα στείλει τη συγκεκριμένη φόρμα στο λογιστήριο της εταιρίας.

The screenshot shows the Bonita User Application interface. On the left, there's a sidebar with 'To do', 'My tasks', and 'Done tasks' sections. The main area is divided into 'Filters' and 'Task list'. The 'Task list' shows a task named 'Examine Expenses' under the 'Company' process. The right panel displays the 'Examine Employee Travel Expenses' form. It includes fields for 'Extra Hotel Cost' (179), 'Extra Transit Cost' (50), 'Extra Airplane Cost' (365), 'Covered By Administration Cost' (595), and 'Final Cost' (595). There's a checkbox for 'Are Expenses Correct' and a 'Submit' button.

Αφού ολοκληρωθεί και αυτό το βήμα ο λογιστής που είναι συνδεδεμένος στον λογαριασμό του θα δει την εντολή πληρωμής από τον υπεύθυνο και θα πρέπει αρχικά να πατήσει το κουμπί take για να πάρει τον έλεγχο του task.

The screenshot shows the Bonita User Application interface. On the left, there's a sidebar with 'To do', 'My tasks', and 'Done tasks' sections. The main area is divided into 'Filters' and 'Task list'. The 'Task list' shows a task named 'Accountant Fill Personal Info' under the 'Company' process. The right panel displays the 'Accountant Personal Info' form. It includes fields for 'Accountant Name', 'Accountant Surname', 'Accountant ID', 'Accountant Email', 'Accountant Phone Number', 'Accountant Home Number', 'Accountant Fax', 'Accountant Address Street', and 'Account Address Number'. There's a 'Submit' button at the bottom.

Έπειτα θα συμπληρώσει τα στοιχεία του και αφού πατήσει το submit θα του εμφανιστεί το ποσό που πρέπει να πληρώσει στον employee ενώ θα πρέπει

κίολας να επιλέξει στην ένδειξη Accountant In Charge τον εαυτό του και να πατήσει το submit.

Bonita User Application Mac Kots

Tasks Cases Processes

Filters

Process **All** Case Case ID

Search...

In task name column

Task list

Take Release

<input type="checkbox"/> Task name	Process name	Due date
<input type="checkbox"/> Forward Payment	Company	-

1 - 1 / 1

Form Comments

Send Payment To Employee

Expenses Received By Employee

Covered By Administration Cost

595

Accountant In Charge

Accountant In Charge

Accountant Name

Mac

Accountant Surname

Kots

Accountant Id

accountant1

Accountant Email

mackots@gmail.com

Accountant Phone Number

6983521688

Accountant Home Number

2105987684

Accountant Fax

2105987684

Τέλος ο employoee λαμβάνει την ειδοποίηση πληρωμής και βλέπει το ποσό που του επέστρεψε η εταιρία καθώς και τα στοιχεία του λογιστή που ανέλαβε την συγκεκριμένη συναλλαγή.

Bonita User Application Eric Gige

Tasks Cases Processes

Filters

Process **All** Case Case ID

Search...

In task name column

Task list

Take Release

<input type="checkbox"/> Task name	Process name	Due date
<input type="checkbox"/> Receive Payment	Company	-

1 - 1 / 1

Form Comments

Receive Payment Notifiacion

Expenses Received By Employee

Covered By Administration Cost

595

Final Cost *

595

☐ Are Expenses Covered

Accountant In Charge

Accountant Name

Mac

Accountant Surname

Kots

Accountant Id *

accountant1

Accountant Email *

mackots@gmail.com

Accountant Phone Number *

6983521688

Accountant Home Number

2105987684

Bonita User Application Eric Gige

Tasks Cases Processes

Filters

Process **All** Case Case ID

Search...

In task name column

Task list

Take Release 1 - 1 / 1

<input type="checkbox"/>	Task name	Process name	Due date
<input type="checkbox"/>	Receive Payment	Company	-

1 - 1 / 1

Form Comments

595

☐ Are Expenses Covered

Accountant In Charge

Accountant Name

Accountant Surname

Accountant Id *

Accountant Email *

Accountant Phone Number *

Accountant Home Number

Accountant Fax

Accountant Address Street

Account Address Number

Submit

Αφού ο employee πατήσει το submit το process έχει πλέον ολοκληρωθεί και τερματίζεται οριστικά.