**Τεχνολογία Λογισμικού**

Class - Diagram v 0.1

**Εικόνα που περιέχει σχεδίαση

Περιγραφή που δημιουργήθηκε αυτόματα**

Γιαννέλη Χριστίνα ΑΜ: 1090055 4o έτος

Καραγεώργος - Γεωργόπουλος Πολύκαρπος ΑΜ: 1051332 9° έτος

Μπαλής Γεώργιος, ΑΜ: 1040996 (παλαιός 235230) 14o έτος

Νεζερίτης Μάριος, ΑΜ: 1080400 5o έτος.

https://github.com/ChristineGi/LeaseIt

**Εικόνα που περιέχει κείμενο, Σχέδιο, διάγραμμα, ορθογώνιο παραλληλόγραμμο

Περιγραφή που δημιουργήθηκε αυτόματα**

**Περιγραφή Κλάσεων**

1. **User:** Οντότητα η οποία αναφέρεται στα στοιχεία των χρηστών του συστήματος.

* Attributes:
  + **+ username: String**
  + **+ password: String**
  + **+ email: String**

1. **Database:** Οντότητα αποθήκευσης πληροφοριών απαραίτητων για την λειτουργικότητα του συστήματος.

* Attributes:
  + **- vehiclePreferencesList: List<VehiclePreferences>**
  + **- userDetailsList: List<UserDetails>**
  + **- emails: List<String>**
  + **- sessionSettings: SessionSettings**
  + **- posts: List<Post>**
  + **- leasingSubscriptions: List<LeasingSubscriptions>**
  + **- logInteractions: List<LogInteractions>**
* Methods:
  + **+ getVehiclePreferences(): List<VehiclePreferences>**
  + **+ searchVehicles(preferences: VehiclePreferences): List<VehicleDetails>**
  + **+ retrieveSessionSettings(): SessionSettings**
  + **+ getPosts(): List<Post>**
  + **+ getLeasingSubscriptions(): List<LeasingSubscriptions>**
  + **+ getLogInteractions(): List<LogInteractions>**
* Constructors:
  + **+ Database()**

1. **Vehicle:** Οντότητα η οποία αναφέρεται στα στοιχεία των οχημάτων.

* Attributes:
  + **- vehicleDetailsList: List<VehicleDetails>**
* Methods:
  + **+ fetchVehicleDetails(vehicleId: String): VehicleDetails**
  + **+ requestVehicleData(): void**
  + **+ getVehicleStatusData(): void**
  + **+ getMaintenanceHistory(): void**
  + **+ searchVehicles(preferences: VehiclePreferences): List<VehicleDetails>**
  + **+ getLeasedVehicles(): List<VehicleDetails>**
  + **+ leaseVehicle(vehicleId: String): void**
* Constructors:
  + **+ Vehicle()**

1. **Leasing Subscriptions**: Ειδικότερη οντότητα Διαχείρισης Συνδρομών Μίσθωσης οχημάτων.

* Attributes:
  + **+ leaseID: String**
  + **+ vehicleID: String**
  + **+ userID: String**
  + **+ status: String**
  + **+ locationConfigured: boolean**
* Methods:
  + None specified in the provided context.
* Constructors:
  + **+ LeasingSubscriptions(leaseID: String, vehicleID: String, userID: String, status: String, locationConfigured: boolean)**

1. **Lease Contract:** Ειδικότερη οντότητα Συμβολαίου Μίσθωσης οχημάτων.

* Attributes:
  + **+ contractID: String**
  + **+ terms: LeasingTerms**
  + **+ subscriptionList: List<LeasingSubscriptions>**
  + **+ active: boolean**
* Methods:
  + **+ calculateLeasingTerms(vehicleDetails: VehicleDetails): LeasingTerms**
  + **+ requestLeasingTerms(calculatedTerms: LeasingTerms): LeasingTerms**
  + **+ finalizeLeaseContract(): void**
  + **+ updateLeaseStatus(status: String): void**
  + **+ checkActiveLease(userID: String): boolean**
  + **+ getVehicleInfo(leaseID: String): VehicleDetails**
  + **+ getUserLeasesByStatus(userID: String, status: String): List<LeasingSubscriptions>**
  + **+ updateSubscription(subscription: LeasingSubscriptions): void**
  + **+ getLeasingSubscription(subscriptionID: String): LeasingSubscriptions**
  + **+ creditRewards(userID: String, amount: double): void**
  + **+ disableAccount(userID: String): void**

1. **Email:** Οντότητα δημιουργίας και διαχείρισης emails.

* Attributes:
  + **- userEmails: Map<String, List<String>>**
  + **- emailServer: String**
  + **- emailPort: int**
* Methods:
  + **+ sendEmail(username: String, message: String): void**
  + **+ detectSessionEnd(): void**
  + **+ sendProofEmail(): void**
  + **+ notifyCompletionEmail(): void**
  + **+ qrCodeEmail(): void**
  + **+ sendAccountLockEmail(userID: String): void**

1. **Message:** Οντότητα διαχείρισης μηνυμάτων τα οποία εμφανίζονται στις οθόνες της εφαρμογής.

**Message**

* **Attributes:**
  + **- messageQueue: List<String>**
  + **- messageTimestamp: Map<String, String>**
* **Methods:**
  + **+ displayRejectMessage(errorMessage: String): void**
  + **+ displaySuccessMessage(successMessage: String): void**
  + **+ showMessage(message: String): void**
  + **+ displayRejectionMessage(errorMessage: String): void**
  + **+ notifyLowScore(userID: String): void**
* **Constructors:**
  + **+ Message()**

1. **Dealership:** Οντότητα η οποία αναφέρεται στα στοιχεία Αντιπροσωπείας Αυτοκινήτων.

* **Attributes:**
  + **None specified in the provided context.**
* **Methods:**
  + **+ retrieveDealerships(): List<String>**
  + **+ checkAvailability(dealershipID: String): List<String>**
  + **+ displayPickupTimes(): void**
  + **+ notifyDelay(): void**
  + **+ monitorPreparation(): void**
  + **+ scheduleDealershipAppointment(time: String): boolean**
  + **+ reportFailedVerification(): void**
* **Constructors:**
  + **+ Dealership()**

1. **Vehicle Leasing:** Οντότητα διαχείρισης διαδικασίας Μίσθωσης Οχήματος.

* **Attributes:**
  + **None specified in the provided context.**
* **Methods:**
  + **+ showLeasingVehicleScreen(): void**
  + **+ submitPreferences(preferences: VehiclePreferences): boolean**
  + **+ selectVehicle(vehicleId: String): void**
  + **+ confirmLeasingTerms(terms: LeasingTerms): void**
  + **+ reviseLeasingApplication(): void**
  + **+ abandonLeasingProcess(): void**
  + **+ proceedToPayment(): void**
  + **+ showRetrievedSettings(): void**
  + **+ returnLeasingOverview(): String**
* **Constructors:**
  + **+ VehicleLeasing(database: Database, vehicle: Vehicle, leaseContract: LeaseContract, taxGateway: TaxGateway, messageService: Message, paymentGateway: PaymentGateway, emailService: Email)**

1. **Vehicle Pickup:** Οντότητα διαχείρισης ραντεβού παραλαβής οχήματος.

* Attributes:
  + None specified in the provided context.
* Methods:
  + **+ showVehiclePickup(): void**
  + **+ promptForSync(): void**
  + **+ selectPickupTime(time: String): void**
  + **+ displayPickupTimes(): void**
  + **+ promptRegenerateCode(): void**
* Constructors:
  + **+ VehiclePickup(database: Database, leaseContract: LeaseContract, googleMaps: GoogleMaps, dealership: Dealership, calendar: Calendar, wallet: Wallet, emailService: Email, userDetails: UserDetails)**

1. **Google Maps:** Οντότητα διασύνδεσης της εφαρμογής με τους χάρτες της Google.

* **Attributes:**
  + **- apiKey: String**
  + **- currentLocation: String**
  + **- destination: String**
* **Methods:**
  + **+ syncLocation(): void**
  + **+ getRoute(): void**
  + **+ locateParkingAndFuel(): void**
  + **+ shareIncidentData(): void**
  + **+ searchEngineeringServices(): void**
  + **+ trackHelperLocation(): void**
  + **+ loadVehicleCoords(): void**
  + **+ loadRouteToVehicle(): void**
  + **+ startNavigation(): void**
  + **+ updateLocation(): void**
  + **+ collectLocationData(): void**
  + **+ noAssistanceAvailable(): void**

1. **Tax Gateway:** Οντότητα διαχείριση Πιστοληπτικής Ικανότητας χρήστη.

* **Attributes:**
  + **None specified in the provided context.**
* **Methods:**
  + **+ checkCreditworthiness(userDetails: UserDetails, terms: LeasingTerms): boolean**
  + **+ returnCreditworthinessResult(): void**
* **Constructors:**
  + **+ TaxGateway()**

1. **Payment Gateway:** Οντότητα διαχείρισης χρηματοοικονομικών συναλλαγών.

* **Attributes:**
  + **None specified in the provided context.**
* **Methods:**
  + **+ completePayment(paymentDetails: PaymentDetails): PaymentConfirmation**
  + **+ paymentConfirmation(transactionId: String): void**
  + **+ processPayment(): void**
  + **+ verifyBankCredentials(): void**
  + **+ authorizationReceived(): void**
  + **+ completeTransaction(): void**
* **Constructors:**
  + **+ PaymentGateway()**

1. **Calendar:** Οντότητα διαχείρισης και διασύνδεσης Calendar του χρήστη.

* **Attributes:**
  + **None specified in the provided context.**
* **Methods:**
  + **+ scheduleUserAppointment(time: String): boolean**
  + **+ syncUserCalendar(appointment: String): void**
* **Constructors:**
  + **+ Calendar()**

1. **QR Code:** Οντότητα δημιουργίας QR κωδικού.

* **Attributes:**
  + **None specified in the provided context.**
* **Methods:**
  + **+ generateQRCode(): String**
  + **+ verifyQRCode(qrCode: String): boolean**
* **Constructors:**
  + **+ QRCode(emailService: Email, userDetails: UserDetails)**

1. **Location Configuration:** Ειδικότερη οντότητα ορισμού γεωγραφικών ορίων οχήματος για παρακολούθηση ασυνήθιστης συμπεριφοράς.

* **Attributes:**
  + **+ active: boolean**
* **Methods:**
  + **+ checkLocationConfiguration(): void**
  + **+ getBoundarySettings(): void**
  + **+ setBoundaries(): void**
* **Constructors:**
  + **+ LocationConfiguration(active: boolean)**

1. **Wallet:** Οντότητα διαχείρισης ψηφιακών wallets (Google, Apple).

* **Attributes:**
  + **None specified in the provided context.**
* **Methods:**
  + **+ importInWallet(qrCode: String): void**
  + **+ connectDigitalWallet(): void**
* **Constructors:**
  + **+ Wallet()**

1. **Cloud:** Οντότητα διασύνδεσης Βάσης Δεδομένων με το Google Cloud Platform.

* **Attributes:**
  + **- locationConfigured: boolean**
* **Methods:**
  + **+ connectToCloud(): boolean**
  + **+ attemptReconnection(): void**
  + **+ loadVehicleCoords(): void**
  + **+ requestVehicleData(): void**
  + **+ collectDrivingData(): void**
  + **+ commitLeasingRenewal(): void**
  + **+ storeVideoCall(): void**
  + **+ getVehicleStatusData(): void**
  + **+ getMaintenanceHistory(): void**
* **Constructors:**
  + **+ Cloud()**

1. **Notification:** Οντότητα διαχείρισης και δημιουργίας ειδοποιήσεων στον χρήστη.

* **Attributes:**
  + **None specified in the provided context.**
* **Methods:**
  + **+ notifyCompletion(): void**
  + **+ showNotification(): void**
  + **+ createNotification(): void**
  + **+ sendPushNotification(): void**
  + **+ pushNotificationToNearbyMembers(): void**
  + **+ notifyHelpAssistance(): void**
* **Constructors:**
  + **+ Notification()**

1. **Vehicle Τracking:** Οντότητα παρακολούθησης θέσης και μοτίβων κίνησης οχήματος.

* Attributes:
  + None specified in the provided context.
* Methods:
  + **+ completeRegistration(): void**
  + **+ requestLocationPermission(): void**
  + **+ resumeTracking(): void**
  + **+ displayData(): void**
  + **+ retryTracking(): void**
  + **+ noUnusualActivityDetected(): void**
  + **+ completeNavigation(): void**
* Constructors:
  + **+ VehicleTracking(cloud: Cloud, googleMaps: GoogleMaps, message: Message, leaseContract: LeaseContract)**

1. **AI Location Analyzer:** Ειδικότερη οντότητα ανάλυσης τοποθεσίας μέσω Τεχνητής Νοημοσύνης.

* Attributes:
  + None specified in the provided context.
* Methods:
  + **+ analyzeRouteLogs(): void**
* Constructors:
  + **+ AILocationAnalyzer()**

1. **Navigation System:** Οντότητα διαχείρισης και λήψης οδηγιών πλοήγησης χρήστη στην τρέχουσα θέση του οχήματος.

* Attributes:
  + None specified in the provided context.
* Methods:
  + **+ loadRouteToVehicle(): void**
  + **+ startNavigation(): void**
  + **+ updateLocation(): void**
  + **+ completeNavigation(): void**
* Constructors:
  + **+ NavigationSystem()**

1. **Vehicle Maintenance:** Γενική οντότητα διαχείρισης συστήματος Συντήρησης Οχημάτων.

* Attributes:
  + None specified in the provided context.
* Methods:
  + **+ showVehicleMaintenance(): void**
  + **+ displayStatus(analysisResults: String): void**
  + **+ displayIssueDescription(): void**
  + **+ displayMaintenanceSchedule(centerDetails: String): void**
* Constructors:
  + **+ VehicleMaintenance(vehicle: Vehicle, ai: AI, cloud: Cloud, maintenanceCenter: MaintenanceCenter, emailService: Email, calendar: Calendar, userDetails: UserDetails, leaseContract: LeaseContract)**

1. **Vehicle Status Data:** Οντότητα διαχείρισης και αποστολής Onboard Diagnostics System (OBD) του οχήματος στο Cloud.
2. **Maintenance History:** Οντότητα διαχείρισης και αποθήκευσης Αρχείων Συντήρησης και Service.
3. **AI Service Analyzer:** Ειδικότερη οντότητα ανάλυσης δεδομένων κατάστασης οχήματος μέσω Τεχνητής Νοημοσύνης για τον εντοπισμό αναγκών συντήρησης.

* **Methods:**
  + **+ analyzeMaintenanceNeeds(): boolean**
  + **+ vehicleStatus(maintenanceNeeds: String): void**
  + **+ enablePeriodicChecks(): void**
  + **+ analyzeRisk(issueDetails: String): void**

1. **Maintenance Centers:** Οντότητα διαχείρισης Εξουσιοδοτημένων Αντιπροσώπων Service.

* Attributes:
  + None specified in the provided context.
* Methods:
  + **+ prioritizeUrgentMaintenance(): void**
  + **+ getAuthorizedCenters(): List<String>**
  + **+ confirmAppointment(centerDetails: String, appointment: String): void**
  + **+ getMaintenanceSchedule(center: String): List<String>**
* Constructors:
  + **+ MaintenanceCenters()**

1. **Urgent Service:** Ειδικότερη οντότητα διαχείρισης επειγόντων ζητημάτων για άμεση επισκευή οχήματος.

* Attributes:
  + None specified in the provided context.
* Methods:
  + **+ displayIssueDescription(): void**
  + **+ submitIssueDescription(): void**
* Constructors:
  + **+ UrgentService()**

1. **Reminder:** Οντότητα δημιουργίας και διαχείρισης υπενθυμίσεων.

**29. Reminder**

* **Attributes:**
  + **- reminderList: List<ReminderDetails>**
  + **- sessionEndDetected: boolean**
* **Methods:**
  + **+ scheduleReminder(reminderDetails: ReminderDetails): void**
  + **+ detectSessionEnd(): void**
  + **+ setReminder(): void**
* **Constructors:**
  + **+ Reminder()**

1. **Leasing Renewal:** Οντότητα διαχείρισης ανανέωσης Μίσθωσης Οχήματος.

* Attributes:
  + **- renewalOptions: List<RenewalPackage>**
  + **- selectedPackage: RenewalPackage**
* Methods:
  + **+ calculateRenewalOptions(): void**
  + **+ displayRenewalOptions(): void**
  + **+ getRenewalPackage(selectedPackage: String): RenewalPackage**
  + **+ displayRenewalSummary(selectedPackage: String): void**
  + **+ generateDigitalAgreement(): void**
  + **+ showDrivingReport(): void**
  + **+ displayImprovementProgress(): void**
  + **+ showLeasingRenewalScreen(): void**
  + **+ signAgreement(): void**
  + **+ commitLeasingRenewal(): void**
  + **+ displayImprovementGoals(): void**

1. **Driving Data Analyzer:** Οντότητα ανάλυσης οδηγικής συμπεριφοράς, δεδομένων OBD και Αστυνομικών Αρχείων για τον υπολογισμό του προφίλ επικινδυνότητας.

* Attributes:
  + **- drivingData: List<DrivingData>**
  + **- riskProfile: RiskProfile**
* Methods:
  + **+ analyzeDrivingData(): void**
  + **+ calculateRiskProfile(): void**
  + **+ notifyLowScore(): void**

1. **Improvement Program:** Οντότητα δημιουργίας για οδηγούς με προφίλ υψηλού κινδύνου προς βελτίωση οδηγικής συμπεριφοράς, με δυνατότητα παρακολούθησης προόδου.

* Attributes:
  + **- programList: List<ProgramDetails>**
* Methods:
  + **+ createImprovementProgram(): void**

1. **Technical Support:** Γενική οντότητα διαχείρισης Συστήματος Υποστήριξης.

**Methods:**

* showSupportScreen()
* enterQuery(category,query)
* exit()
* provideFeedback()
* approveAgentSupport()

1. **Chatbot:** Ειδικότερη οντότητα διαχείρισης ΑΙ Chatbot υποστήριξης.

* **Attributes:**
  + **- active: boolean**
  + **- supportOffered: boolean**
* **Methods:**
  + **+ activateChatbot(): void**
  + **+ offerAgentSupport(): void**

1. **Query Processor:** Οντότητα επεξεργασίας ερωτημάτων υποστήριξης με έμφαση στις λέξεις – κλειδιά.

* Attributes:
  + **- queryHistory: List<String>**
  + **- insufficientSolutions: List<String>**
* Methods:
  + **+ processQuery(query: String): void**
  + **+ notifyInsufficientSolution(): void**
  + **+ getAnswers(answers: List<String>): void**
  + **+ trainModel(video: String, feedback: String): void**
  + **+ modelFeedback(): void**

1. **Agent Support:** Ειδικότερη οντότητα αίτησης ανθρώπινης υποστήριξης.

* Attributes:
  + **- supportAgent: String**
  + **- requestQueue: List<SupportRequest>**
* Methods:
  + **+ routeToSupportAgent(): void**

1. **Feedback:** Οντότητα διαχείρισης ανατροφοδότησης μετά την παροχή υποστήριξης.

* **Attributes:**
  + **- feedbackList: List<FeedbackDetails>**
* **Methods:**
  + **+ processFeedback(): void**
* **Constructors:**
  + **+ Feedback()**

1. **Chat Session:** Οντότητα διαχείρισης και συλλογής συνομιλίας και αλληλεπίδρασης.

* **Attributes:**
  + **- chatHistory: List<ChatMessage>**
  + **- sessionID: String**
* **Methods:**
  + **+ createNewChatSession(): void**
  + **+ displayAnswers(): void**
  + **+ displaySelectedSolution(): void**
  + **+ revisedAnswer(): void**
  + **+ getChatSessionData(): void**

1. **Video Call:** Οντότητα διαχείρισης video call για Technical Support.

* Attributes:
  + **- callHistory: List<VideoCallDetails>**
  + **- callID: String**
* Methods:
  + **+ initiateVideoCall(): void**
  + **+ recordVideoCall(): void**
* Constructors:
  + **+ VideoCall()**

1. **Community:** Γενική οντότητα διαχείρισης περιεχομένου Κοινότητας.

* Attributes:
  + **- communityContent: List<Content>**
  + **- filteredPosts: List<Post>**
* Methods:
  + **+ showCommunityScreen(): void**
  + **+ updateCommunityContent(): void**
  + **+ displayFilteredPosts(): void**
  + **+ displayPosts(): void**
  + **+ selectFilters(): void**
  + **+ interactWithPosts(): void**
* Constructors:
  + **+ Community()**

1. **Content Management System:** Οντότητα διαχείρισης και προβολής περιεχομένου που δημιουργείται από τους χρήστες στο φόρουμ της κοινότητας.

* Attributes:
  + **- postList: List<Post>**
  + **- contentForms: List<ContentForm>**
* Methods:
  + **+ requestPosts(): void**
  + **+ generateContentForm(): void**
  + **+ applyFilters(): void**
  + **+ revisePostContent(): void**
  + **+ getPostID(): String**
  + **+ exitCommunity(): void**

1. **AI Profile Analyzer:** Ειδική οντότητα Μηχανικής Μάθησης για την ανάλυση του ιστορικού αναζήτησης και των αλληλεπιδράσεων χρήστη για προσαρμοσμένο περιεχόμενο.

* **Attributes:**
  + **- userInteractions: List<Interaction>**
* **Methods:**
  + **+ analyzeUserInteractions(): void**

1. **Post:** Οντότητα δημιουργίας αναρτήσεων στην οντότητα Community.

* Attributes:
  + **- postData: String**
  + **- interactions: List<Interaction>**
* Methods:
  + **+ createPostData(postData: String): void**
  + **+ filterPosts(): void**
  + **+ updatePostInteractions(): void**

1. **Content Moderator:** Οντότητα NLP μοντέλου ανάλυσης και συμμόρφωσης των αναρτήσεων με βάση τα πρότυπα του Community.

* Attributes:
  + **- contentData: String**
* Methods:
  + **+ scanContent(contentData: String): void**

1. **Refer And Earn:** Γενική Οντότητα διαχείρισης ανταμοιβών και επιβραβεύσεων.

* Attributes:
  + **- rewardList: List<Reward>**
* Methods:
  + **+ showReferAndEarn(): void**
  + **+ displayAccountLocked(): void**
  + **+ shareLink(): void**
  + **+ confirmContacts(contacts: List<String>): void**
  + **+ displayInstructions(): void**
  + **+ displayRewards(): void**

1. **Link Generator:** Οντότητα δημιουργίας συνδέσμου παραπομπής.

* Attributes:
  + **- referralLink: String**
* Methods:
  + **+ generateReferralLink(): void**
  + **+ displayLink(): void**
  + **+ displayInstructions(): void**

1. **Reward System:** Οντότητα διαχείρισης ανταμοιβών και εκπτώσεων βάση προϋποθέσεων εταιρείας.

* Attributes:
  + **- rewardList: List<Reward>**
* Methods:
  + **+ detectLinkExpired(): void**
  + **+ detectNewRegistration(): void**
  + **+ updateAccountWithRewards(): void**
  + **+ creditRewards(): void**
  + **+ activateRewardProcess(): void**

1. **ΑΙ Activity Analyzer:** Οντότητα ανάλυσης δραστηριότητας χρηστών για πιθανή κατάχρηση της υπηρεσίας Refer And Earn.

* **Attributes:**
  + **- activityData: List<Activity>**
* **Methods:**
  + **+ checkForMisuse(): void**
  + **+ suspendLink(): void**

1. **Contacts:** Οντότητα διασύνδεσης εφαρμογής με τις επαφές του χρήστη.

* **Attributes:**
  + **- contactList: List<Contact>**
* **Methods:**
  + **+ accessContacts(): void**
  + **+ selectContacts(): void**

1. **SMS Gateway:** Οντότητα διαχείρισης και αποστολής SMS.

* **Attributes:**
  + **- sentMessages: List<SMS>**
  + **- accountLockMessages: List<SMS>**
* **Methods:**
  + **+ prepareSMS(referralLink: String, contacts: List<String>): void**
  + **+ sendSMS(): void**
  + **+ sendAccountLockSMS(): void**
  + **+ sendSMSUpdate(): void**
  + **+ generateIncidentSMS(): void**

1. **New Customer Detector:** Οντότητα εντοπισμού νέων πελατών μέσω συνδέσμων παραπομπής.

* Attributes:
  + **- scheduler: Scheduler**
  + **- newRegistrations: List<Customer>**
* Methods:
  + **+ setScheduler(): void**
  + **+ detectNewRegistration(): void**

1. **Incident Report:** Γενική οντότητα διαχείρισης δήλωσης περιστατικού – ατυχήματος.

* **Attributes:**
  + **- incidentForm: IncidentForm**
  + **- helperDetails: HelperDetails**
* **Methods:**
  + **+ displayIncidentForm(): void**
  + **+ assistanceArrival(): void**
  + **+ selectContactOption(): void**
  + **+ confirmHelperAssistance(): void**
  + **+ displayResults(): void**
  + **+ showHelperDetails(): void**
  + **+ logEngineeringInfo(): void**
  + **+ enableIncidentServiceSystem(): void**

1. **Incident Service System:** Γενική οντότητα διαχείρισης ατυχήματος.

* Attributes:
  + **- repairTime: int**
* Methods:
  + **+ estimateRepairTime(): void**
  + **+ logEngineeringInfo(): void**
  + **+ activateRewardProcess(): void**

1. **Roadside Assistance:** Οντότητα διαχείρισης παρόχων Οδικής Βοήθειας.

* Attributes:
  + **- assistanceRequests: List<AssistanceRequest>**
  + **- repairEstimates: List<RepairEstimate>**
* Methods:
  + **+ sendAssistanceRequest(): void**
  + **+ noAssistanceAvailable(): void**
  + **+ cancelDefaultAssistance(): void**
  + **+ estimateRepairTime(): void**

1. **Geofencing Service:** Οντότητα διαχείρισης Geofencing τεχνολογίας για εντοπισμό χρηστών ενός συγκεκριμένων γεωγραφικών ορίων.

* Attributes:
  + **- geofencingSettings: GeofencingSettings**
* Methods:
  + **+ activateGeofencing(): void**

1. **ΑΙ Incident Image Analyzer:** Οντότητα ανάλυσης μέσω Τεχνητής Νοημοσύνης εικόνων Ατυχήματος προς εκτίμηση ζημιάς και χρόνου επισκευής.

* Attributes:
  + **- incidentImages: List<IncidentImage>**
* Methods:
  + **+ analyzeIncidentForm(): void**

1. **Government Gateway:** Οντότητα συλλογής Πιστοληπτικής Ικανότητας χρήστη.

* Attributes:
  + **- policeRecords: List<PoliceRecord>**
* Methods:
  + **+ getPoliceRecords(): void**

1. **UI:** Γενική οντότητα διαχείρισης των οθονών της εφαρμογής.