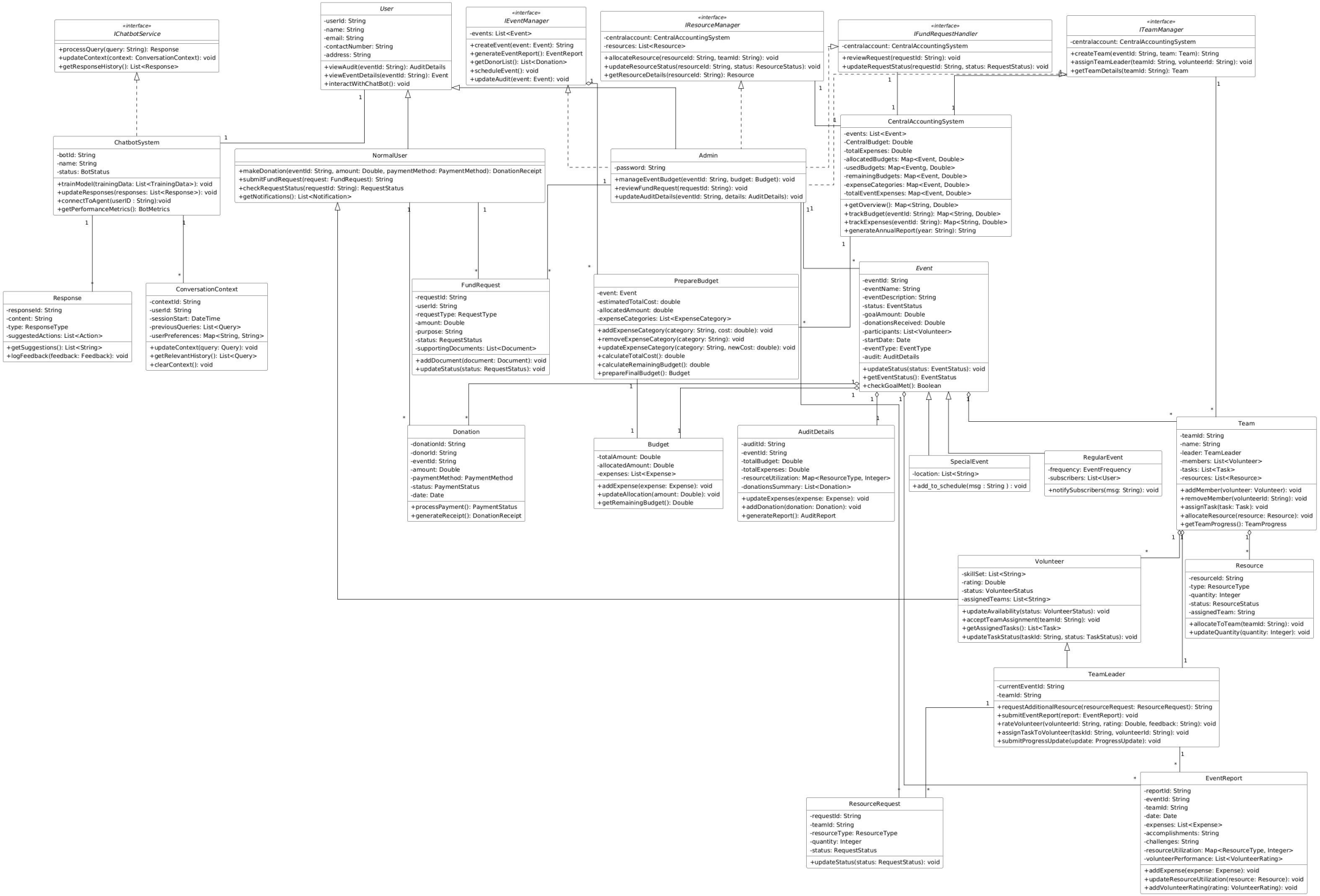
«interface»
IChatbotService
+processQuery(query: String): Response
+updateContext(context: ConversationContext): void
+getResponseHistory(): List<Response>
User
-userId: String
-name: String
-email: String
-contactNumber: String
-address: String
+viewAudit(eventId: String): AuditDetails
+viewEventDetails(eventId: String): Event
+interactWithChatBot(): void
«interface»
IEventManager
-events: List<Event>
+createEvent(event: Event): String
+generateEventReport(): EventReport
+getDonorList(): List<Donation>
+scheduleEvent(): void
+updateAudit(event: Event): void
«interface»
IResourceManager
-centralaccount: CentralAccountingSystem
-resources: List<Resource>
+allocateResource(resourceId: String, teamId: String): void
+updateResourceStatus(resourceId: String, status: ResourceStatus): void
+getResourceDetails(resourceId: String): Resource
«interface»
IFundRequestHandler
-centralaccount: CentralAccountingSystem
+reviewRequest(requestId: String): void
+updateRequestStatus(requestId: String, status: RequestStatus): void
«interface»
ITeamManager
-centralaccount: CentralAccountingSystem
+createTeam(eventId: String, team: Team): String
+assignTeamLeader(teamId: String, volunteerId: String): void
+getTeamDetails(teamId: String): Team
ChatbotSystem
-botId: String
-name: String
-status: BotStatus
+trainModel(trainingData: List<TrainingData>): void
+updateResponses(responses: List<Response>): void
+connectToAgent(userID : String):void
+getPerformanceMetrics(): BotMetrics
NormalUser
+makeDonation(eventId: String, amount: Double, paymentMethod: PaymentMethod): DonationReceipt
+submitFundRequest(request: FundRequest): String
+checkRequestStatus(requestId: String): RequestStatus
+getNotifications(): List<Notification>
Admin
-password: String
+manageEventBudget(eventId: String, budget: Budget): void
+reviewFundRequest(requestId: String): void
+updateAuditDetails(eventId: String, details: AuditDetails): void
CentralAccountingSystem
-events: List<Event>
-CentralBudget: Double
-totalExpenses: Double
-allocatedBudgets: Map<Event, Double>
-usedBudgets: Map<Eventg, Double>
-remainingBudgets: Map<Event, Double>
-expenseCategories: Map<Event, Double>
-totalEventExpenses: Map<Event, Double>
+getOverview(): Map<String, Double>
+trackBudget(eventId: String): Map<String, Double>
+trackExpenses(eventId: String): Map<String, Double>
+generateAnnualReport(year: String): String
Response
-responseId: String
-content: String
-type: ResponseType
-suggestedActions: List<Action>
+getSuggestions(): List<String>
+logFeedback(feedback: Feedback): void
ConversationContext
-contextId: String
-userId: String
-sessionStart: DateTime
-previousQueries: List<Query>
-userPreferences: Map<String, String>
+updateContext(query: Query): void
+getRelevantHistory(): List<Query>
+clearContext(): void
FundRequest
-requestId: String
-userId: String
-requestType: RequestType
-amount: Double
-purpose: String
-status: RequestStatus
-supportingDocuments: List<Document>
+addDocument(document: Document): void
+updateStatus(status: RequestStatus): void
PrepareBudget
-event: Event
-estimatedTotalCost: double
-allocatedAmount: double
-expenseCategories: List<ExpenseCategory>
+addExpenseCategory(category: String, cost: double): void
+removeExpenseCategory(category: String): void
+updateExpenseCategory(category: String, newCost: double): void
+calculateTotalCost(): double
+calculateRemainingBudget(): double
+prepareFinalBudget(): Budget
Event
-eventId: String
-eventName: String
-eventDescription: String
-status: EventStatus
-goalAmount: Double
-donationsReceived: Double
-participants: List<Volunteer>
-startDate: Date
-eventType: EventType
-audit: AuditDetails
+updateStatus(status: EventStatus): void
+getEventStatus(): EventStatus
+checkGoalMet(): Boolean
Donation
-donationId: String
-donorId: String
-eventId: String
-amount: Double
-paymentMethod: PaymentMethod
-status: PaymentStatus
-date: Date
+processPayment(): PaymentStatus
+generateReceipt(): DonationReceipt
Budget
-totalAmount: Double
-allocatedAmount: Double
-expenses: List<Expense>
+addExpense(expense: Expense): void
+updateAllocation(amount: Double): void
+getRemainingBudget(): Double
AuditDetails
-auditId: String
-eventId: String
-totalBudget: Double
-totalExpenses: Double
-resourceUtilization: Map<ResourceType, Integer>
-donationsSummary: List<Donation>
+updateExpenses(expense: Expense): void
+addDonation(donation: Donation): void
+generateReport(): AuditReport
SpecialEvent
-location: List<String>
+add_to_schedule(msg : String) : void
RegularEvent
-frequency: EventFrequency
-subscribers: List<User>
+notifySubscribers(msg: String): void
Team
-teamId: String
-name: String
-leader: TeamLeader
-members: List<Volunteer>
-tasks: List<Task>
-resources: List<Resource>
+addMember(volunteer: Volunteer): void
+removeMember(volunteerId: String): void
+assignTask(task: Task): void
+allocateResource(resource: Resource): void
+getTeamProgress(): TeamProgress
Volunteer
-skillSet: List<String>
-rating: Double
-status: VolunteerStatus
-assignedTeams: List<String>
+updateAvailability(status: VolunteerStatus): void
+acceptTeamAssignment(teamId: String): void
+getAssignedTasks(): List<Task>
+updateTaskStatus(taskId: String, status: TaskStatus): void
Resource
-resourceId: String
-type: ResourceType
-quantity: Integer
-status: ResourceStatus
-assignedTeam: String
+allocateToTeam(teamId: String): void
+updateQuantity(quantity: Integer): void
TeamLeader
-currentEventId: String
-teamId: String
+requestAdditionalResource(resourceRequest: ResourceRequest): String
+submitEventReport(report: EventReport): void
+rateVolunteer(volunteerId: String, rating: Double, feedback: String): void
+assignTaskToVolunteer(taskId: String, volunteerId: String): void
+submitProgressUpdate(update: ProgressUpdate): void
EventReport
-reportId: String
-eventId: String
-teamId: String
-date: Date
-expenses: List<Expense>
-accomplishments: String
-challenges: String
-resourceUtilization: Map<ResourceType, Integer>
-volunteerPerformance: List<VolunteerRating>
+addExpense(expense: Expense): void
+updateResourceUtilization(resource: Resource): void
+addVolunteerRating(rating: VolunteerRating): void
ResourceRequest
-requestId: String
-teamId: String
-resourceType: ResourceType
-quantity: Integer
-status: RequestStatus
+updateStatus(status: RequestStatus): void