

Message
+ messageText: string + targetContact: Contact
+ init(messageText: string, targetContact: Contact) {} + createNewMessage(messageText: string) {} + chooseFromSampleMessage(messageText: string) {}

MessageTemplates
+ messageList: [string] (premade list of sample messages )
+ addCustomMessage(message: string)

ScheduleMessage
+ messageToSend: Message + checkTime: Bool + checkLocation: Bool + timeSchedule: NSTime + locatinSchedule: String/Map?
+ Schedule(contact: Contact, messageText: Message, type: Time/Location) {} + setTime() + setGeofence( ) (Address/Pin) - use set radius

Contact
+ firstName:string + lastName:string + phoneNumber:int - contactID:int
+ init( firstName: string, lastName: string, phoneNumber: int, contactID: int) {}

ContactList
+ contactArray:[Contact] - count:Int { return contactArray.count}
+ init(contactArray: [Contact]) {} + addContact(contact: Contact) { contactArray.append(contact) }

ContactGroup
+ contactGroupArray:[Contact] - count:Int { return contactGroupArray.count} - contacts: [int] { return contactGroupArray.numbers}
+ init(contactGroupArray: [Contact]) {} + addContactToGroup(contact: Contact) { contactGroupArray.append(contact) }

SwiftJSON
Same as shown in class

GiantBombService
+ api_key: string
+ init( api_key: string) {} + parseJSONResponse() + interactAPI() + createMap()

Settings