

API url: <https://botapi.rubika.ir> - API version: 2

Request header: token

JSON input parameters: api_version, method, data

JSON output parameters: status, status_det, data

Server Side:

Method: sendMessage

Input data: chat_id, text, bot_keypad (Keypad object), reply_timeout (in seconds, 0 means no timeout), file_id, rnd, location (Location object)

Output data: message_id

Method: requestUploadFile (less than 2MB)

Input data: type (Image, File), file_name

Output data: file_id, upload_url, hash_send_file

Method: sendFile (post to upload_url)

Headers: file-id, hash-send-file, bot-token

Body: File

Method: getChatMessages

Input data: chat_id, start_message_id, limit

Output data: messages (array of Message objects)

Method: getMessages

Input data: start_message_id, limit

Output data: messages (array of Message objects)

Method: getChatInfo

Input data: chat_id

Output data: first_name, last_name, username, last_message_id

Method: updateBotEndpoints

Input data: type (NewMessage/Selection/SearchSelection), url

Output data: status (Done, InvalidUrl)

Method: paymentRequest

Input data: chat_id, order_id, payment_options (amount (in Toman))

Output data: payment_token

Method: getPaymentStatus

Input data: order_id, chat_id, payment_token

Output data: status (Settled, NotSettled, Reversed, NotPaid, Error, NotUsed)

Method: settlePayment

Input data: order_id, chat_id, payment_token

Output data: status (Done, Reject)

Method: reversePayment

Input data: order_id, chat_id, payment_token

Output data: status (Done, Reject)

Bot Side:

Method: receiveMessage

Input data: message (Message object), type (Message, API)

Output data: bot_keypad (Keypad object), text_message

Method: getSelection

Input data: chat_id, selection_id, first_index, last_index

Output data: items (array of ButtonSimple objects)

Method: searchSelection

Input data: chat_id, selection_id, search_text, limit

Output data: items (array of ButtonSimple objects)

Objects:

Message: message_id, chat_id, text, file_inline (FileInline object), reply_to_message_id, time, aux_data (AuxData object), location (Location object)

FileInline: file_url, file_name, size (in Bytes), width, height, duration

Keypad: rows (array of KeypadRow objects), auto_button_id (optional)

KeypadRow: buttons (array of Button objects)

Button: id, type (Simple, Selection, Calendar, NumberPicker, StringPicker, Payment, CameraImage, CameraVideo, GalleryImage, GalleryVideo, File, Audio, RecordAudio, Alert, Textbox, Call, SMS, Barcode, Location, AskMyPhoneNumber, AskMyLocation), button_view (ButtonSimple Object), button_selection (ButtonSelection object), button_calendar (ButtonCalendar object), button_number_picker (ButtonNumberPicker object), button_textbox (ButtonTextbox object), button_call (ButtonCall object), button_sms (ButtonSMS object), button_string_picker (ButtonStringPicker object), button_payment (button_payment_token), button_alert (ButtonAlert object), reply_type (Message, API), button_location (ButtonLocation object)

ButtonSimple: text, image_url, type (TextOnly, TextImgThu, TextImgBig)

ButtonSelection: selection_id, search_type (None, Local, Api), get_type (Local, Api), items (array of ButtonSimple objects), is_multi_selection, columns_count

ButtonCalendar: default_value (could be null), type (DatePersian, DateGregorian), min_year, max_year

ButtonNumberPicker: min_value, max_value, default_value (could be null)

ButtonStringPicker: items (array of strings), default_value

ButtonAlert: message

ButtonLocation: default_pointer_location (Location object), default_map_location (Location object)

ButtonCall: phone_number

ButtonSMS: to_number, message

ButtonTextbox: type_line (SingleLine, MultiLine), type_keypad (String, Number)

AuxData: start_id, button_id, order_id, order_status (Check, Error)

Location: longitude, latitude