

Telegram Road

A platform for development of chatbots

Release 1.3

Telegram Road is a platform for creation and development of chatbots

What is a chatbot?

A chatbot is an interactive application, in which the user chats with a program. Usually the program asks questions, the user answers, and next questions depend on the given answers.

How does the user chat?

Interaction with Telegram Road can occur either through :

- a computer (desktop, notebook, tablet...) presenting dialogs on a screen, and on which the user replies with keyboard and mouse;
- a smartphone, on which the application **Telegram** has been installed

New technologies of interaction are currently under development.

What is Telegram?

Telegram is a messaging application, like Whatsapp or Viber. Its peculiarity, beyond a strong accent on security, is the possibility to message not only from one-to-one or one-to-groups, but also one-to-large groups (channels) and one-to app , that is, a **bot**

Does Telegram run on smartphones only ?

No, Telegram can also be installed on a desktop (Windows, Mac) or run as a web application (on the main *browsers*).

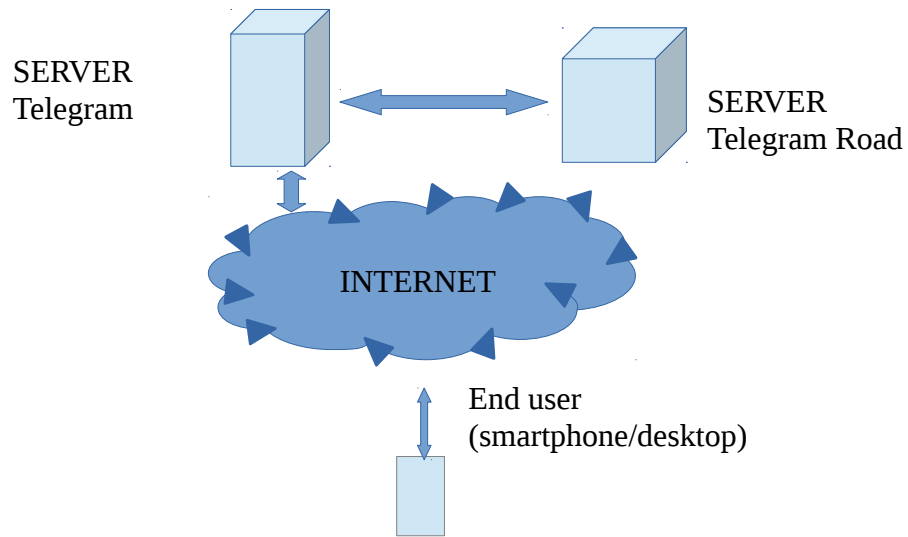
What is Telegram Road for?

Telegram Road offers companies, agencies, associations the possibility to automate the interaction with their customers/users, leveraging a well-known technology (messaging on a smartphone) and leaving great freedom in the building of their own application (in the following, “app”).

I can't understand the difference between Telegram and Telegram Road

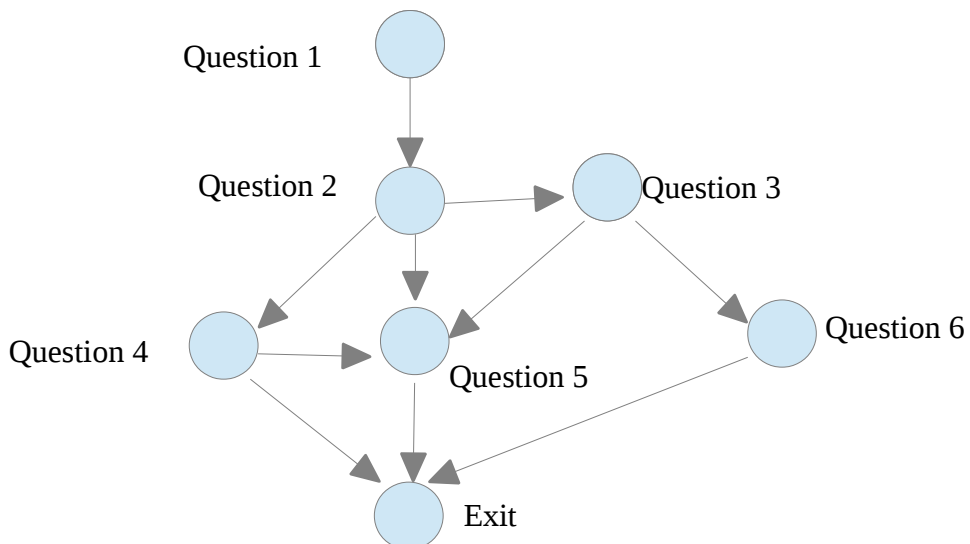
Telegram Road offers the user applications (chatbots) which are perceived by the user as Telegram bots. Telegram is the only thing to be installed on the telephone (or computer). **That's a cornerstone of the Telegram Road approach, opposite to spread of dedicated Apps, which lead to clogging of smartphones due to their hunger of hardware resources.**

Here is the architecture of the Telegram Road service. Chatbots run on the Telegram Road server, but are seen by the end user through Telegram, running on his/her own smartphone or desktop, with no difference.



How can I build an application on Telegram Road?

1 – First of all you need paper and pencil. On a sheet, you can sketch the questions, drawn as boxes (or “balls”) which are connected through arcs , representing answers, and leading to other boxes. You can draw a graph. representing your application, like this:



This graph will undergo modifications and updates, as long as the App will grow. So, don't worry if the correct form is not found at once.

2 – The next step is translating it on a bot on Telegram. You need to have Telegram on your own

mobile device, or on your PC (*desktop*). In any case, the address to visit is :

<https://telegram.org/dl>

Note: since the installation is tied to a cell phone number, if you give the same number you are granted to have on any platform perfect replicas of your profile (chat, bot, preferences, ...), always aligned in real time. The cell phone number is only required in this passage , and by no means will it be necessary or released to whatsoever.

3 – After Telegram installation, you will search for a particular bot, called **BotFather**, which acts as a “host” for creation of new bots. The creation of a new bot, as well as all other Telegram functions, is **free of charge**.

4 – With the command **/newbot**, after a handful of easy passages, we can give our bot a name and get an unique key (the *token*) giving access to the newborn bot .

5 – The next step , always within Telegram, is looking for another bot, **Troadbot**. This bot, peculiar for the platform Telegram Road, will ask us the name to be given to our new bot (better the same of Telegram) and the token which BotFather assigned it. (Best idea is to copy/paste).

6 – After registering the new bot, we are given the possibility to run a form-based application where we will be able to input the graph drawn in step 1, working in the so called *backstage* of Telegram Road.

All these activities are up to the administrator of the chatbot (the “App manager”), who owns credentials (password) granting him control on the Apps.

3 – At this point the App is ready to be released on a desktop or tablet platform (via web), and possibly undergo a debug phase. For instance, this is how our chatbot questions set might be available to the manager:

Domande

Domande :

- 58 (S) Offerta -> 58 Offerta
- 59 (S) Competenze -> 59 Competenze
- 60 (S) Attuazione -> 60 Attuazione
- 217 (S) Presentation -> 54 Presentazione
- 218 (S) Offer -> 218 Offer
- 219 (S) Skills -> 59 Competenze
- 220 (S) Timing -> 60 Attuazione
- 104 (S) Conclusione -> 54 Presentazione
- 221 (S) Conclusion -> 54 Presentazione
- 54 (S) Presentazione -> 54 Presentazione

Nuova domanda

Edita domanda

Cancella domanda

Prova la domanda

Risposte :

- Base IT -> 219 (S) Skills
- Office automation... -> 219 (S) Skills
- Advanced tools... -> 219 (S) Skills
- Advanced Internet... -> 219 (S) Skills

Nuova risposta

Edita risposta

Cancella risposta

Esci

7 – We will insert here the boxes , corresponding to the questions, and the arcs, corresponding to the answers , our App is composed of.

8- At any time, the App can be as well tested on our mobile device, running it in Telegram environment

That's all!

What kind of applications lend themselves to Telegram Road?

All applications reflecting a cascading of question/answers, maybe complex. These are typically static applications, whose structure (the graph) does not change, but the variable part is the way the end user can cross the graph. Typical applications can be:

- skill tests
- surveys
- *troubleshooting* to support diagnostic on devices, appliances , machines, ...

and much more.

Another class of applications suitable to Telegram Road implementation are **dynamic applications**, for which the behaviour depends on data which are specific for the running user/session.

Examples of dynamic applications might be:

- databases
- agendas
- “decision support systems” , expert systems based on user's experience

Dynamic applications offer users a spectrum of wider and more powerful solutions, with proportional complexity as well.

Are there demos?

At current time, here are the available demo apps:

@Monthdays : demo giving the number of days of a month, considering bissextile years

@EsameGuidabot : survey to gather preferences for a new IT course

@TRoadClip_bot : clipboard to enter memos to find out quickly

@troadmusic_bot : database for everything about music: artists, records, instruments, and much more!

To try a demo, search in Telegram the bot with its name , e.g. . @Monthdays, and accept the activation. Relax, nothing will be installed.

How can I get more info?

E-mail : info@telegramroad.com

Web : <http://telegramroad.com>

Can I create an App of my own?

Yes, and it is free of charge for static Apps.

Are there further developments?

Yes, Telegram Road sets challenging goals, not only as development tool for **static applications**, but also for **dynamic applications**, for which the behaviour of the chatbot depends on data structures growing and breeding on the run.

In detail, it is possible to build chatbots on:

- dynamic app, static databases, in which the database architecture does not vary. For example, one can build an Address Book to gather contacts, to be queried and edited through the questions of the chatbot
- dynamic app, dynamic mode databases, in which to store whatever info, following everyone's specific needs. The power of Telegram Road platforms stems on the possibility to build flexible data structures , able to bend to the end user's needs.

As these functions are way more complex, they are object of a specific doc (Manager Manual).

Have fun!

Expanding an app

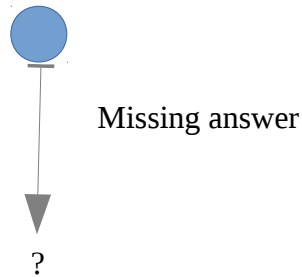
As from release 1.3, not only a manager is given the possibility to build an app, but also the end user can. That corresponds to a collaborative way to create an app, as anybody can add nodes to a new or an existing app, bringing new options (nodes, arcs) to a graph representing the app.

This can easily bring, for instance, to :

- troubleshooting or “how-to” manuals, in which anyone knowing the right way to solve an issue is given the possibility to share his/her knowledge (a *wiki*)
- tourism guides, where new hints, news and material are added on the way by tourists themselves
- games in which a story is actually bred by the very same gamers in a collaborative fashion

This list can naturally grow with the fantasy of users and developers.

When a new app is born, the very first question carries a “**missing answer**” option.



This is a keystone of a Telegram Road application, as the user choosing this option:

- will be asked the answer (text) he feels missing (this text will replace the “missing answer” string)
- will be asked if he knows a follow-up for such answer. This can be thought of as asking a way to solve a problem, which is pointed by the new answer being added
- if the user does not know the way, answering “no” will let him close this option, waiting for someone else to follow. Adding an answer will just let other users know that an issue is still “open” and waiting for a solution.
- When another user (or, maybe, the user himself) will be visiting the app and following the new answer, he will be asked if it feels ready to submit a solution.
- Accepting, he will be asked a new question, that is a node in the graph, along with all its features, such as an introduction, the question itself and possibly a picture to be shown.
- The newborn question is given the features “open” = true and “public”=false. This will allow further possibility of growth of the graph, but inhibit the publication of the node until an administrator has authorized it. Only the creator will be allowed to see at the moment.
- New arcs (answers) will be possibly added to the new questions, repeating the steps above. Each new node will carry along a “missing answer” option to let it grow.

This possibility of feeding contents from the end user side gives Telegram Road a great power to build innovative and collaborative applications with no developing tools.