



# National Institute Of Technology, Delhi

## SOFTWARE DEVELOPMENT PROJECT

**"ONLINE CHATBOT"**

**COMPUTER SCIENCE ENGINEERING(CSE)**



**SUBMITTED BY**

**AKASH KUMAR**

**191210007**

**&**

**ARUSHI JAISWAL**

**191210013**

**SUBMITTED TO**

**MS. SONIYA SHARMA**

**DEPARTMENT OF COMPUTER SCIENCE( UG)**

**NATIONAL INSTITUTE OF TECHNOLOGY, DELHI**

## **ABSTRACT:**

A chatterbot or Chatbot aims to make a conversation between both human and machine. The machine has been embedded with knowledge to identify the sentences and making a decision itself as response to answer a question. The response principle is matching the input sentence from user. The present technical project consist of developing an expert System for college enquiry desk using an android based Chabot, through Artificial Intelligence technology and virtual assistance (Human-machine conversation),transmitting natural language to a server.

## **FIRST WE GATHER THE INFORMATION BY LOOKING INTO DIFFERENT ASPECTS OF THE CUSTOMER PROBLEM.**

1. Who's the target customer?
2. What is the Chatbot's overall goal, Basically the outline of the basic features of the Chatbot.
3. What are the current solutions that the customer uses?
4. What would the Chatbot do better in order to replace it with the current solution?
5. The reliability of such systems.
6. As the Chatbot relies on the content feed to it,how will you provide the meaningful data
7. according to the customers need?
8. Will the Chatbot provide fast on-boarding as it is a complex software?
9. Even though it is a complex software does it have an easy to use interface on both sides the customer as well as the company.

## **QUESTIONS ASKED BY THE STAKEHOLDERS:-**

1. What will be different in a year after the project is completed?
2. In what ways do you see the project helping you achieve your mission.
3. What happens if the project is not completed before the expected deadline?
4. What problems do you see this project solving with evidence?
5. What do you think could go wrong with this project?
6. What are the major risks to the project success?
7. What concerns you the most about this software?
8. When the software is commercialized ?
9. What are the expenses related to its hosting and other requirements needed to make it commercialized?
10. The advantages he/she foresees after the implementation of the software.

## **TENTATIVE LIST OF STACKHOLDERS ARE:-**

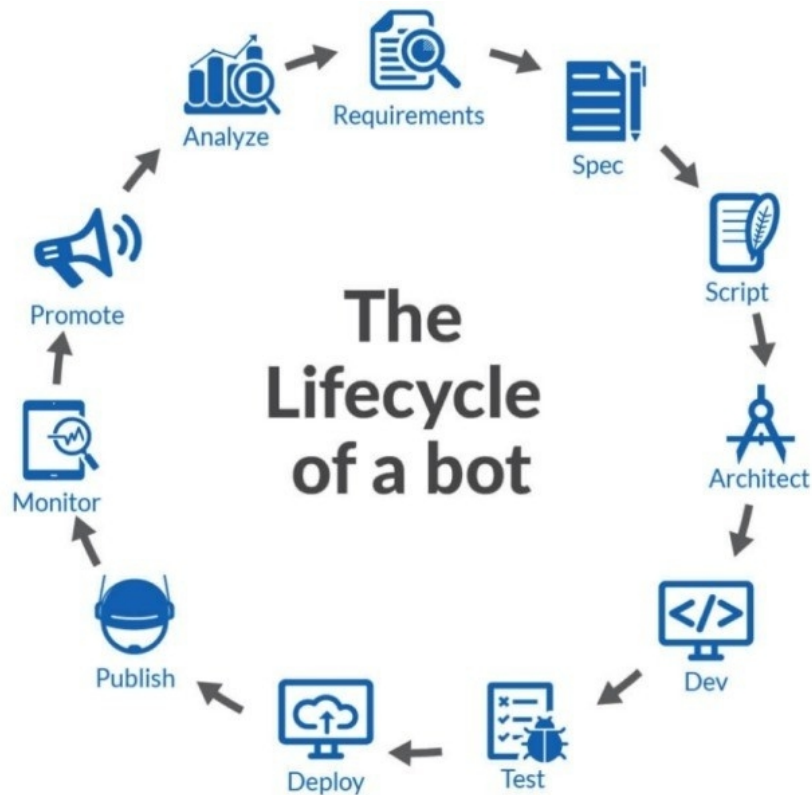
- Project leader

- Senior management
- Project Team Member
- Project customer
- Resources Manager
- Line Managers
- Product user group
- Project testers
- Any group impacted by the project as it progresses
- Any group impacted by the project when it is Completed
- Subcontractors to the project
- Consultants to the project

## **AFTER GOING THROUGH ALL THE REQUIREMENTS OF THE STAKEHOLDER AND THE CUSTOMER WE DEFINED OUT PROBLEM STATEMENT AS FOLLOWS:**

Artificial intelligence chatbot is a technology that makes interactions between man and machines using natural language possible. chatbot are functions like a typical search engine. Although chatbot just produced only one output instead of multiple outputs/results, the basic process flow is the same where each time an input is entered, the new search will be done. Nothing related to previous output. This research is focused on enabling chatbot to become a search engine that can process the next search with the relation to the previous search output. In chatbot context, this functionality will enhance the capability of chatbot's input processing.

## . PLANNING:-



## REQUIREMENTS:

Gather market requirements for the Chatbot , As we have defined the problems above.  
Requirements

## SPEC:

Develop a product spec for the bot identifying the features and functionality of the bot. The features should deliver the benefits identified in the Requirements step. Note that the spec must also include short and long description of the bot along with other collateral that will be required later in the Publish step.

## 3. SCRIPT:

While the first two steps are similar to that of other software lifecycles, this step is unique to the bot building process. While websites and apps have structured interfaces, bots have

a conversational interface. Instead of building wireframes like we do for websites and apps, this involves building conversational scripts that represent user interactions.

The conversational script must be representative of actual user conversations. Since the conversational interface has no tabs or prompts, the bot script must guide the user towards accomplishing the desired task. Depending on the context, the script may or may not support NLP capabilities. If the bot is designed with NLP in mind, the bot script must handle a wide range of variations in user inputs. Note that NLP and AI must be used with caution. Too much of it can raise user expectations and lead to disappointment.

#### **4. ARCHITECT:**

Create the engineering design for the bot. This includes both the front-end and back-end components. The front-end refers to the conversational interface — translating user input into specific actions and vice versa. The back-end refers to computation performed by the bot as well as integrations to other web services.

#### **5. DEV:**

This is the development stage where the bot is developed. Given the conversational interface, bot developers will find themselves iterating a lot more between coding and testing than in traditional software development. Soon as the bot is coded to handle a specific set of conversational statements, it's a good practice to unit-test the code through the messaging interface.

Developers must also insert tracking probes into the bot — these will be helpful at the Track stage.

#### **6. TEST:**

As described above, the testing is deeply intertwined with the development process. However, testing is tricky for a bot developer. The code must be tested not just in the emulator, but also in the actual messaging platform. Given the diversity of messaging apps, and the differences in message rendering, this can be a time-consuming process. Also, different messaging platforms have different guidelines and access limitations for test bots.

Apart from the unit-testing during the dev cycle, this step also includes the final QA of the bot. The QA process must run the bot through the conversational scripts developed above. The QA process also needs to be aware of, and ensure compliance with, the Publish guidelines of the messaging platform. Messaging platforms have different guidelines (e.g. bots must not spam, they must introduce themselves, explain themselves, behave themselves, handle exceptions etc).

#### **7. DEPLOY:**

Once the bot is built, it must be deployed to a hosted environment. The hosted environment must be stable and needs its own monitoring and devops support.

## **8. PUBLISH:**

Once the bot is tested and deployed, it must now be submitted to the various app stores for approval. Each messaging platform has a different approval process, with varying degrees of rigor. The submissions require a variety of descriptive elements such as a short description, a long description, images, scripts, videos etc (best to develop these in the Spec phase). Messaging platforms also require their bots to be “well behaved” and self-explanatory. This process can take from days to weeks and involve multiple iterations, with multiple gotchas for the first time publisher. Making this harder is the fact that approval processes are evolving too, making this a moving target!

## **9. MONITOR:**

Once the bot is published, it must be monitored. I don’t mean just ops monitoring as mentioned in the Deploy phase. I mean monitoring the bot using actual conversational scripts. The ops monitoring may indicate systems are well, but the bot may still be unresponsive to certain user conversations. The best way to monitor the bot is to monitor the user conversation.

## **10. PROMOTE**

While publishing to the bot store is the first step, there are other options to make your bot discoverable. There are cross-platform bot stores that are off the messaging platforms that may still drive traffic to your bot. New advertising channels are emerging that will introduce your bot to new users.

## **11. ANALYZE:**

As the bot starts being used, its performance must be tracked and results analyzed. This involves reviewing both conversation logs and usage metrics. Analyzing conversational behavior is different from analyzing click-paths for websites and apps. The ideal conversation is the one that enables users to accomplish their task with the least effort. Depending on context, this may mean shorter or longer conversations — the one that imposes the least cognitive load on the user is the best. Developers must identify and fix the leakage points — where the conversations are breaking or being abandoned.

## **REPEAT:**

The learnings from the Analyze phase can be cycled back into the bot development process to build an ever-improving bot. Some bots may even be programmed with self-learning AI programs that need inputs from users and trainers to continuously improve themselves.

# **POSSIBLE RISKS INVOLVED:-**

## **INFLEXIBLE:**

Chatbots are designed for specific responses and try to find out what the user needs, so they could solve it. However, there could be instances when they themselves don't know what the solution or required information is. They can't do more than what they've been programmed for. They can't reply to those questions which they are not taught. This can cause frustration among customers when they don't get a satisfactory response.

## **COSTLY:**

If you are trying to setup and automate a lengthy or complex process, it will take a lot of time, money and effort. The maintenance required would also be extensive. This can make it less cost effective if a human could have been trained with less cost and effort. Also the requirement could be for one-off customer or a small amount of customers as compared to the costs associated. Hence it is always better to conduct a Cost-Benefit Analysis before starting anything.

## **LOSS OF SERVICE:**

If your system stops working or the functionality breaks, and your interface goes offline, you have no backup in position to deal with such situation. There could be loss of customers and business until you get the system back online. Also, the Human Team behind the Chatbots could be working well over their capacity and this could cost your company even more.

## **USER DISLIKE:**

Some clients especially the older generation would prefer to talk to a real person. They may not have the knowledge of how to interact with the Chatbot. So even if they Chatbot is doing its Job perfectly even then the customer would get annoyed. Hence it is better to conduct a demographic research before setting up AI or Human Based Team.

## **EMOTIONLESS:**

Its work like a robot, they can't feel or pickup human emotions. They can't sense the tone of the customer or what the customer is thinking during the conversation. This can cause frustration among customers even when the Chatbot is doing its Job effectively.

## **TIME CONSUMING SYSTEM:**

Where there are great number of processes and corresponding responses, it will take more time to develop the system and begin your operations. This can prove to be costlier and delays your business plans to start your operations. You need to determine what responses need to be added or changed and all these calls for a longer settling period.



**THANK YOU**

