TEAM ID	PNT2022TMID01145
PROJECT NAME	AI Based Discourse for Banking Industry

1. INTRODUCTION:

Recent technological developments have transformed the way consumers and financial institutions interact with each other (Shaikh and Karjaluoto, 2015). Moreover, the COVID-19 pandemic has led to rapid shift to digital technologies and banks have transitioned to remote sales and provision at a fast pace (McKinsey, 2020).

The rise of artificial intelligence (AI) based technology is contributing extensively to this transformation as more and more banks have begun to implement AI-based applications with the aim of deepening customer relationships, providing more personalized offers, detecting and preventing fraud, improving processes for anti-money laundering, and cost saving (Business Insider, 2021).

One very popular and impactful form of technological development in the financial sector is the implementation of the AI-based chatbot technology (Richad et al., 2019). Chatbot technology is a system based on AI that communicates with users and performs basic tasks through chat or speech interfaces (Nguyen and Sidorova, 2017). According to forecast, the chatbot market size is projected to reach 102.29 billion USD by 2025 (Mordor Intelligence, 2019), and the success rate of bot interactions (queries completed without the interference of a human operator) in the banking sector is expected to be over 90% by 2022 (Juniper Research, 2020). Yet, in order for both banks and consumers to exploit from the advantages of the banking chatbot technology, it is important to examine the drivers of consumers' willingness to adopt this technology. The adoption of innovations applied in the banking industry, such as i-banking and m-banking, is widely studied in the literature.

Although the above mentioned studies provide essential contribution to technology adoption in the context of the banking industry, there is limited knowledge on the acceptance of banking chatbots. There are already a few studies that highlighted the importance of studying chatbot acceptance in the context of the banking industry, but only one has conducted a research related to the technology acceptance model (Richad et al., 2019), and one is related to the customer experience for consumers who are banking with chatbot technology (Trivedi, 2019). In order to extend the knowledge regarding chatbot acceptance in banking, the present study aims to investigate the determinants of chatbot adoption in the banking industry, based on technology adoption literature in the banking industry.

The aim of the current study is to identify those factors that have an influence on consumers' intention to use chatbot technology applied in the banking industry. A special adoption model was developed for banking chatbots, extending the TAM model. Data is collected through an online, self-administrated questionnaire and the conceptual model is examined using the PLS-SEM method.

2.1. Project Review:

The project is about a chatbot that using AI (Artificial Intelligence) for Banking industry. This Chatbot use the AI to recognize the user input (Queries) and the chat bot will respond Quickly and give the best solution for the queries raised by the user.

2.2. Purpose:

The Purpose of the Chatbot is reduce the time of the user to clear the queries and Problem.

Chatbots in banking are incredibly powerful and can manage smart communications on behalf of the bank. With a banking bot, it's possible to handle millions of users simultaneously and enhance their experience.

- **Lead Generation and Customer Acquisition**

  Chatbots have no match when interaction comes to play. They can interact with customers for the first time and understand their needs and sentiments behind the conversation.

- **Automated customer support**

  65% of the customer queries are repetitive and can be easily solved by conversational banking chatbots and the rest can be solved with artificial intelligence, and agent support. 65% of the customer queries are repetitive and can be easily solved by conversational banking chatbots and the rest can be solved with artificial intelligence, and agent support.

- **Financial Advisor**

  Banking chatbots have all the data to predict the spending habits of the customers and help them keep their finances on track. With the help of artificial intelligence, banks can provide financial advice to their customers

- **Smart Payment Processing**

  Banks are using chatbots for processing payments as per the instructions. This saves a lot of time for the customers by fetching all the data and processing the payments quickly. Chatbots also help customers by reminding them of the due payments or bills. Removing the redundancy of entering details, again and again, chatbots help customers in sending as well as receiving money proactively and with 100% security.

- **Reviews and feedback**

  Customers would love to give feedback and reviews if their hard-earned money is taken care of by the banks. These reviews can be collected by the banking chatbots. Instead of using the long survey forms, banks can now integrate chatbots on their websites and apps for collecting feedbacks and reviews.

Our Chatbot:

In our project that can do some short of things that can reduce the time and getting information quickly. The chatbot that use the third party API and the user can ask the queries. Our chat bot can give clear solution for the raised problem or queries from the user.

2. LITERATURE SURVEY

With our idea we are going to create a bot. The bot should be able to guide a customer to create a bank account and should it be able to answer the loan queries. Our project is capable of user free text that can recognize the raised question and the proper solution will be given to the user.

2.1. Existing Problem:

- Connections go down, people speak in unique manners, and sometimes the chatbots can't understand them, whether text-based or voice. Goodie, a provider of live-chat platforms, found in a survey of 1,000 banking customers that 60% have an unfavourable view of chatbots and don't trust them.
- While chatbots provide many benefits to organizations in the banking industry, there are some drawbacks. Considering each of these can help you find the best customer service solution for your business.

Here are some of the most common drawbacks of using chatbots for banking.

- *Questions must be programmed beforehand*
  When customers use chatbots, they need to ask questions in specific ways to be able   to get accurate answers.
  Chatbots can only answer questions that have been programmed previously. This downside however, can be eliminated by connecting an effective live agent solution to your CX strategy and adding unresolved questions to your chatbot's knowledge base.

- *Impersonal*
  Most Chatbots use structured flows to provide answers. This can make the experience robotic and feel less personal.
  Answers that are 100% scripted, don't allow flexibility when it comes to regionalisms and different ways of asking questions.
   Plus, chatbots lack the authenticity that a human can provide, so your customers may not feel like their needs were met or that their complaints were heard.

- *Must Keep Information Up-To-Date*
  Chatbots without artificial intelligence can only provide your customers with answers to questions that they already know.

  It is up to you to input new information for chatbots to use.

If you don't keep this information up-to-date, your chatbot may be sending incorrect messages to your customers.

- *Technology Issues*

There can also be some technical issues when it comes to using chatbots for banking.

First, chatbots require your customers to use the internet. If you have customers that do not have access to the internet or are unsure of how to use an online platform, it may not be an ideal customer service solution.

People need to be more tech-savvy to use a chatbot than to make a simple phone call or interact with a customer service representative in person.

Finally, there are other common mistakes that people may make which can make it difficult for chatbots to function correctly. For example, a chatbot without conversational IA will have trouble understanding misspelled words and therefore will not be able to provide the right answer.

- Needs Additional Measures to Protect Identities

  Finally, using a chatbot for your banking services may require additional measures to protect the identities of your users.
  This is because they may be sharing private or sensitive account information.

You will have to find ways to confirm that your user is actually who they say they are. This will prevent you from sending sensitive information to the wrong people.

2.2 Reference:

1. Ryoji Kashiwagi, "Utilization of artificial intelligence in finance"(2005).

2. Geeta Narula, Rakhi Narula,"The Impact of Chat-Bots on the BankingExperience" (2021).

3. Mehmet Ates," Artificial Intelligence In Banking - A Case Study About TheIntroduction Of A Virtual Assistant Into Customer Service"(2017).

4. Andrew No ,"What artificial intelligence can do and can't do rightnow" (2016).

5. Mohamed Hussain Thowfeek,"Drivers Of Artificial Intelligence In BankingService Sectors"(2020).

6. Vinod Kumar Shukla, Sasha Fathima Suhel, Sonali Vyas, Ved Prakash Mishra," Conversation to Automation in Banking Through Chatbot Using Artificial Machine Intelligence LanguageMethodology"(2020).

2.3 Problems Statement:

- Nowadays, banks are not able to resolve the queries of customers atall times related to the products or services in satisfactory way.

- So, in order to overcome the issues which are faced by the customers/people accessing the banks, an intelligent system has to beintroduced to provide people with the best solution-possible.

- The users are bank customers who need a service, available 24/7, toclear all their queries and guide them through the various banking processes, in the language-customers can understand.

- So, an enhanced and smarter way of interaction with the customers has to be built to ensure efficient delivery of service. In order to overcome the user satisfaction issues associated with banking services, a chatbot has to be introduced which will provide personaland efficient communication between the user and the bank.

- It will act as an overall virtual assistant that can facilitate customers toask banking- related questions without visiting the bank or contacting customer services and provide them with relevant-suggestions.

PROBLEM	DESCRIPTION
1. Who Does the problem affects?	Illiterate or old people. People who are new to banking environment. People who don't have time to visit the bank.
2. What are the boundaries of the problem?	Customers who have queries related to banking or trying to use various services of the bank. Customers who want to access or know about the services provided by the bank in their understandable language.
3. What is the issue?	Customers need to visit banks frequently for simple queries. Banks are unable to answer huge volumes of queries efficiently. Not Provide Banking services in a customized manner based on their interests and activities.

4. When does the issues occurs?	When an individual is a busy worker, who has less time and when an individual has no one to help him find a good loan amount and some policies related to the bank. When the customer is unable to visit the bank. When the customer is unaware about the language spoken by the bank employee.
5. Where does the issues occurs?	The issues occur in Commercial banks, Small finance banks, Co-operative banks, Payments banks.
6. Why its important to solve the issues?	The person can easily find about the banking methods & the time can be saved. This helps people who are really in need of loan amounts and any banking services& those unaware of the language can also have access to the services.

3. IDEATION AND PROPOSED SOLUTION:

3.1. Empathy Map Canvas:

An **empathy map** is a widely-used visualization tool within the field of UX and HCI practice. In relation to empathetic design, the primary purpose of an empathy map is to bridge the understanding of the end user. The traditional empathy map begins with four categories: says, thinks, does, and feels. At the center of the map, a user or persona is displayed to remind practitioners and stakeholders what type of individual this research is centered around.

- **Says** category contains what the user says out loud during research or testing. Ideally, each point is written down as close to the user's original words as possible.

- **Thinks** category contains what the user is thinking. While content may overlap with the *Says* category, *Thinks* category exists to capture thoughts users may not want to share willing due to social factors, such as self-consciousness or politeness.

- **Does** category contains the user's action and behavior. This contains what the user is physically doing and captures what actions users are taking.

- **Feels** category contains the user's emotional state in context with their experience. This typically contains information or phrases as to how they feel about the experience.

- **See** category contains information users observed through eyes. It could be what users see in the marketplace or in the immediate environment, other people's saying and doing, or the content they watch or read.

- **Hear** category is what user hears and how that impacts the user. It could be personal connections as well as other recourse such as media. Instead of documenting superficial information streams, team should focus on details that influence the user.
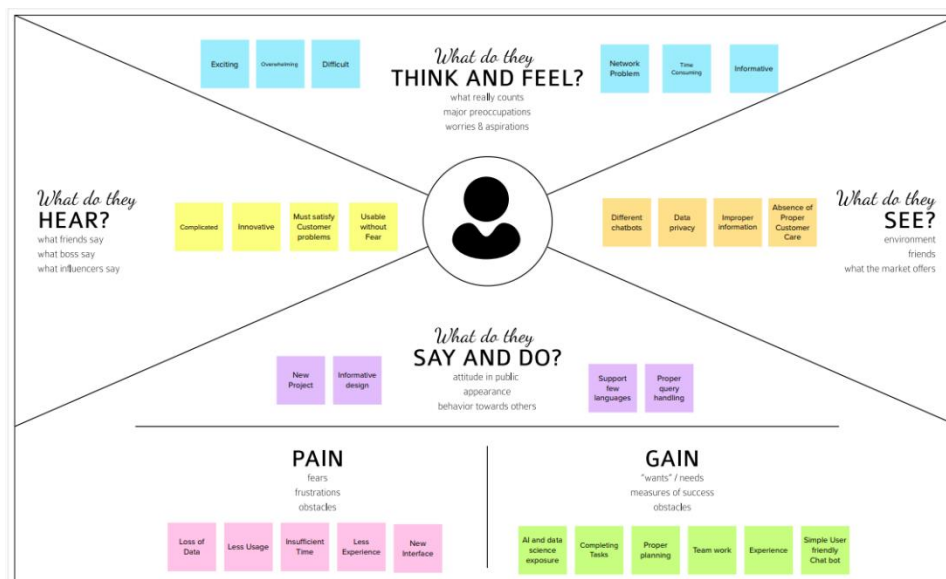
3.2. Ideation and Brain Storming:

1. Define your problem statement
What problem are you trying to solve? Frame your problem as a How Might We statement. This will be the focus of your brainstorm.

How might we build a multilingual chat bot to benefit banking industry.

2. Brainstorm

Write down any ideas that come to mind that address your problem statement.

Brainstorming is a group problem-solving method that involves the spontaneous contribution of creative ideas and solutions. This technique requires intensive, freewheeling discussion in which every member of the group is encouraged to think aloud and suggest as many ideas as possible based on their diverse knowledge. Brainstorming combines an informal approach to problem-solving with lateral thinking, which is a method for developing new concepts to solve problems by looking at them in innovative ways. Some of these ideas can be built into original, creative solutions to a problem, while others can generate additional ideas.

Every team member should give their ideas as a suggestion and they can add their ideas in the Brainstorm. in those collections of ideas, we can select the best ideas and we can work on that and we can achieve our best in that region.

The common ideas will be taken for the next step of the ideation process , there the ideas will be evaluated and give the top priority for the ideas and we can implement the ideas in the form of the priority ranges.

Brainstorm

Write down any ideas that come to mind that address your problem statement.

⏱ 10 minutes

SANJU K JACOB

Accept different Languages	Voice over input Output	User Friendly
Consistency in answers	Personalized Services	Change of Themes

SARABESH KANNAA T M

Fast Response	End to End Encryption	Giving examples for input
Connected with bank staffs	Store the login credentials	Simple Interface

SAKTHIVEL P

Full screen Window	OTP Enable	Fetching History
Solutions through Mails	Giving simple solutions	Better Scalability

VIRUTHIK KUMARESAN S

Customer Satisfaction	Less Process Time	Suggestion and feedback
Available 24/7	Using keywords to help customers	Reduce workload
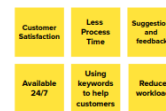
3. Group Ideas

Take turns sharing your ideas while clustering similar or related notes as you go. In the last 10 minutes, give each cluster a sentence-like label. If a cluster is bigger than six sticky notes, try and see if you and break it up into smaller sub-groups.

Group ideas

Take turns sharing your ideas while clustering similar or related notes as you go.
In the last 10 minutes, give each cluster a sentence-like label. If a cluster is bigger than six sticky notes, try and see if you and break it up into smaller sub-groups.

⏱ 20 minutes

Accept Different Language

user Friendly

Reduce Workload

Time Saving

Chat Bot

Fast Response

Better Scalability

customer satisfaction

Easy to Access

4. Prioritize

Your team should all be on the same page about what's important moving forward. Place your ideas on this grid to determine which ideas are important and which are feasible.
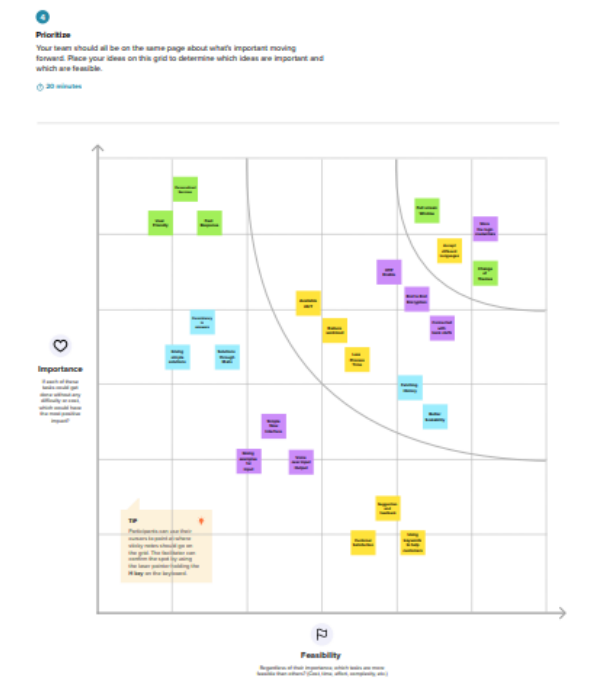
3.3. Proposed Solution:

Proposed Solution means the technical solution to be provided by the Implementation agency in response to the requirements and the objectives of the Project. Having hooked your audience into the problem, now you want to paint a picture of what the world will be like when you solve the problem.

Your proposed solution should relate the current situation to a desired result and describe the benefits that will accrue when the desired result is achieved. So, begin your proposed solution by briefly describing this desired result.

Serial No.	Parameter	Description
1.	Problem Statement (Problem to be solved)	Nowadays, banks are not able to resolve the queries of customers at all times related to the products or services in satisfactory way. To overcome this, an intelligent system has to be introduced to provide people with the best solution possible. The users are bank customers who need a service, available 24/7, to clear all their queries and guide them through the various banking processes, in the language customers can understand. So, an enhanced and smarter way of interaction with the customers has to be built to ensure efficient delivery of service. In order to overcome the user satisfaction issues associated with banking services, a chatbot has to be introduced which will provide personal and efficient communication between the user and the bank. It will act as an overall virtual assistant that can facilitate customers to ask banking- related questions without visiting the bank or contacting customer services and provide them with relevant suggestions.
2.	Idea / Solution description	By using AI techniques, we have to create a chatbot which can solve the user queries in the banking system. We are creating a Watson Assistant and creating Skills in Watson Assistant. Use of NLP techniques like Entities, Intents, Dialogues and then we are deploying skills to generate a preview link.
3.	Novelty / Uniqueness	We have developed a chatbot that works for all scenarios and thus enormous customized test cases are being provided to the chatbot so that it can handle any type of situation. Also, various real world user queries and FAQs of bank users are collected in real time and analysis is done and chatbot is trained correspondingly.
4.	Social Impact / Customer Satisfaction	Customer Queries, Services are solved successfully. Available 24/7. Ease of users to provide a hassle-free internet banking experience. Providing customizable valuable experience to the users. Multi lingual chatbot for different customers.
5.	Business Model (Revenue Model)	With the help of a chatbot the bank is provides its users with a hassle-free internet banking experience and thus increasing customer retention. The cost that occurs on appointing a person is reduced when chatbot is used to handle these queries.
6.	Scalability of the Solution	Since it is a chatbot application, it can be scaled to any extent with the help of adding certain additional modules and test cases. It reflects the Deep and broad perspectives on the bank's global features. Chatbots understands the customer concerns and assists them, round the clock.

3.4. Problem Solution Fit:

The Problem-Solution Fit simply means that you have found a problem with your customer and that the solution you have realized for it actually solves the customer's problem.

Project Title: AI Based Discourse for Banking Industry — Project Design Phase-I - Solution Fit Template — Team ID: PNT2022TMID01145

4. REQUIREMENT ANALYSIS:

Requirements analysis or requirements engineering is a process used to determine the needs and expectations of a new product. It involves frequent communication with the **stakeholders** and end-users of the product to define expectations, resolve conflicts, and document all the key requirements.

The main type of requirement analysis includes functional, and non-functional requirements

4.1 Functional Requirements:

Functional Requirements define what a product must do, what its features and functions are.

They are product features or functions that developers must implement to enable users to accomplish

their tasks. Generally, functional requirements describe system behaviour under specific conditions. For example:

- The system sends a confirmation email when a new user account is created.

- The system sends an approval request after the user enters persona/ information.

FR No.	Functional Requirement (Epic)	Sub Requirement (Story/Sub-Task)
FR-1	Account Creation	• Personal Details Form • Types of Account information
FR-2	General Query Support	• Mail to Customer Care Executive • Bank details • Helpline Number
FR-3	Existing User Support	• Mistakes Correction Query support • Account Freeze action and Security services. • Changes made confirmation through mail or OTP.
FR-4	Load Related Service	• Types of Loan Details. • Interest and Benefit Schemes.
FR-5	Online Banking Support	• UPI linkage to account. • Security Services in unauthorised UPI linkage. • Account Balance Check

4.2 Non-functional Requirements:

Non-functional Requirements, not related to the system functionality, rather define how the system should perform.

They serve as constraints or restrictions on the design of the system across the different backlogs. Here, we'll just briefly describe the most typical non-functional requirements that suits the project.

| FR No.	Non-Functional Requirement	Description
NFR-1	**Usability**: Usability defines how difficult it will be for a user to learn and operate the system.	Customer can access chatbot more efficiently and in a simpler way. Top chat topics are displayed for easy access.
NFR-2	**Security**: Security requirements ensure that the software is protected from unauthorized access to the system and its stored data.	Customer can have utmost security of their information. The details are stored in cloud where the bank employee have total control.
NFR-3	**Reliability**: Reliability defines how likely it is for the software to work without failure for a given period.	If the criteria or the topic which customer expects is not met via chatbot, bank employee will be able to provide details for that issue within a short span of time.
NFR-4	**Performance**: Performance is a quality attribute that describes the responsiveness of the system to various user interactions with it	Chatbot can provide consistency and frequent updating of queries are made without any loss in information.
NFR-5	**Availability:** Availability is gauged by the period that the system's functionality and services are available for use with all operations.	It is available 24x7 and the progress is not lost, even if the servers go down
NFR-6	**Scalability:** Scalability requirements describe how the system must grow without negative influence on its performance.	It can support wide range of users queries and provide instant responses. The queries of more than 1000 people can be answered using the chatbot.

5. PROJECT DESIGN

5.1 Data Flow Diagrams:

A Data Flow Diagram (DFD) is a traditional visual representation of the information flows within a system. A neat and clear DFD can depict the right amount of the system requirement graphically. It shows how data enters and leaves the system, what changes the information, and where data is stored. The four components of data flow diagrams are:

An external entity can represent a human, system or subsystem. It is where certain data comes from or goes to. It is external to the system we study, in terms of the business process. For this reason, people used to draw external entities on the edge of a diagram.

A process is a business activity or function where the manipulation and transformation of data take place. A process can be decomposed to a finer level of details, for representing how data is being processed within the process.

A data store represents the storage of persistent data required and/or produced by the process. Here are some examples of data stores: membership forms, database tables, etc.

A data flow represents the flow of information, with its direction represented by an arrowhead that shows at the end(s) of flow connector.
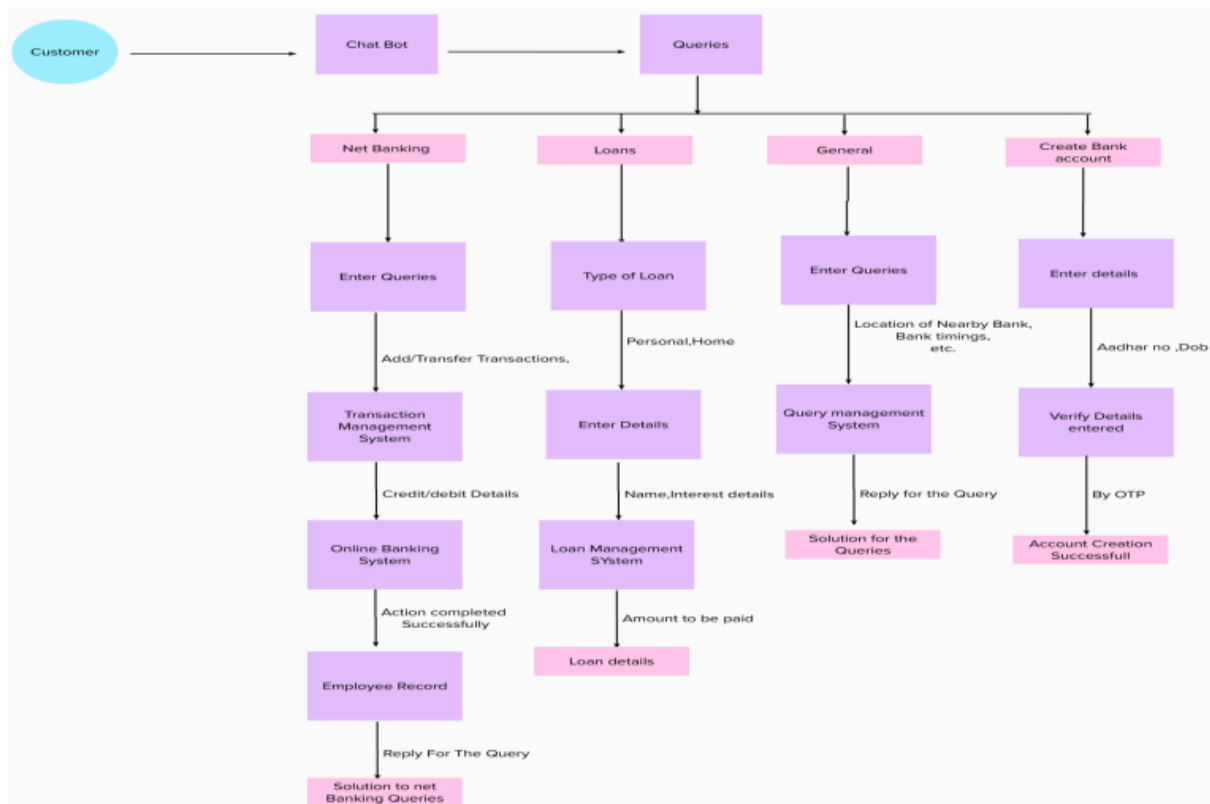
5.2 SOLUTION &TECHNICAL ARCHITECTURE

Technical Architecture:

Technical Architecture (TA) is a form of IT architecture that is used to design computer systems. It involves the development of a technical blueprint with regard to the arrangement, interaction, and interdependence of all elements so that system-relevant requirements are met.

***Choosing an architecture will determine how you deal with performance, fault tolerance, scalability, and reliability.***

A useful architecture diagram has a combination of these three components:

- **Standardized process** flow of information, e.g., top-down reading — this indicates how the components interact with each other.

- **Provides sufficient information in components with logical groupings** — this indicates where constraints lie, e.g., network boundaries.

- **Includes annotations with more information** — steps with slightly more details to facilitate the implementation of solutions, e.g., process description.

A technical architecture diagram provides a bird's eye view of the infrastructure of your organisation. The diagram illustrates how components in a system interact with one another in the large scale of things.

There are two main benefits of an architecture diagram:

1. **They help with understanding** — provides an overview of the available systems and interaction, which facilitates impact assessment from changes easily.

2. **They improve communication and collaboration** — aligns implementation plan across projects and stakeholders to reduce gaps in communication. A useful architecture diagram should address each of the stakeholder's needs to a certain extent.
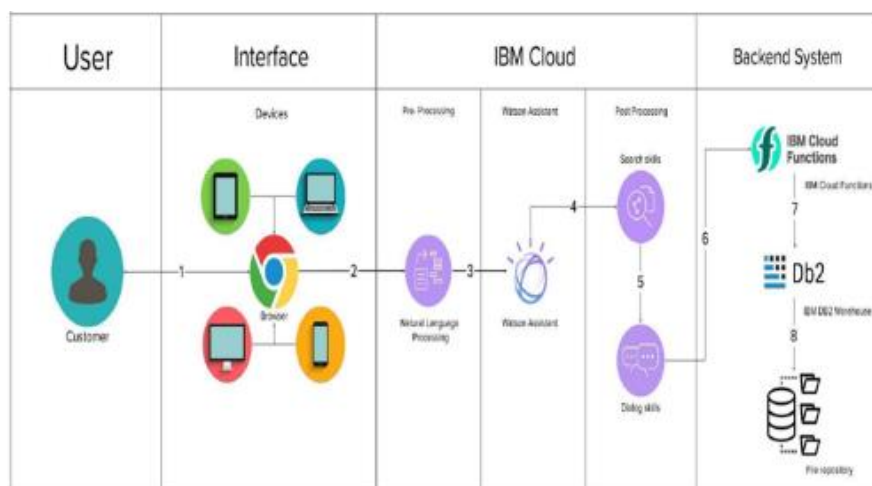
Solution Architecture:

Solution architecture is a complex process – with many sub-processes – that bridges the gap between business problems and technology solutions.

Its goals are to:

- Find the best tech solution to solve existing business problems.
- Describe the structure, characteristics, behaviour, and other aspects of the software to project stakeholders.
- Define features, development phases, and solution requirements.
- Provide specifications according to which the solution is defined, managed, and delivered.
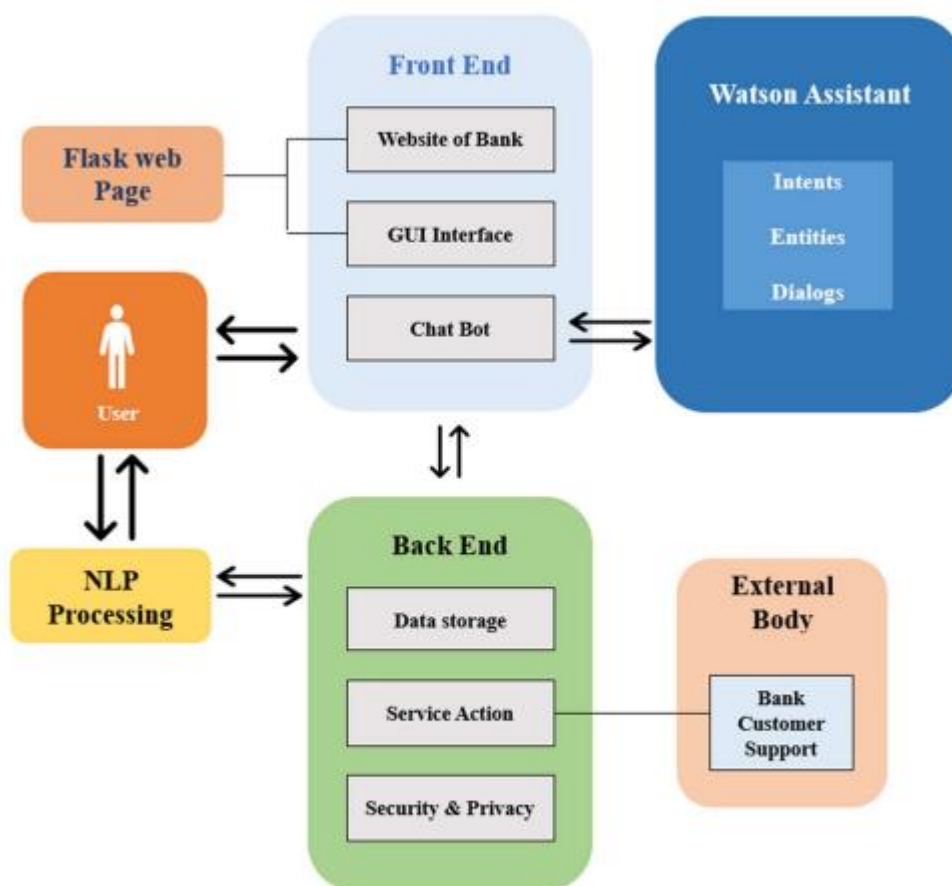
5.3 User Stories

A user story is the smallest unit of work in an agile framework. It's not a feature, but an end goal that the user has when using the software. The user story will convey what the user wants to achieve and states it in a simple, non-technical way. In Agile projects, user stories are organized in a backlog, which is an ordered list of product functions.

"A good user story should be Independent, Negotiable, Valuable, Estimable, Small and Testable."

There are 4 steps to creating good user stories, which require the collaboration of the client and the business analyst on the project,

1. Validate the Needs of the Users

2. Create Epics

3. Writing User Stories

4. Defining Acceptance Criteria

A **good user story** should be:

Independent: Developers should be able to implement the user stories in any sequence

Negotiable: Stories should avoid too much detail and be kept flexible

Valuable: Users get some value from the story

Estimable: The team can use them to plan project timelines

Small: Smaller user stories are easier to estimate than larger ones

Testable: Everyone should be able to document the "definition of done" (how we know the user story is done) and the acceptance criteria.

5.4 Customer Journey Map:

Journey mapping helps you visualize how customers experience your product or service, and how they feel along the way. One of the significant reasons teams use customer journey maps is to understand better how the customer experiences their product. Mapping out customer journeys helps explain why customers make their choices and which aspect of your product is most valuable to them.

Mapping the customer journey has a host of benefits such as:

- Allowing you to optimize the customer onboarding process

- Benchmarking the customer experience desired by your customers against what they receive

- Understanding the differences in buyer personas as they move from prospect to conversion

- through the buying funnel.

- Creating a logical order to your buyer journey.

However, the biggest benefit is simply understanding your customers more. The better you understand their expectations, the more you can tailor the customer experience to their needs.

6. PROJECT PLANNING & SCHEDULING

6.1 Sprint Planning & Estimation

A. Sprint Planning:

The purpose of sprint planning is to define what can be delivered in the sprint and how that work will be achieved. Sprint planning is done in collaboration with the whole scrum team.

A successful session will yield two important strategic items:

1. *The sprint goal:* A short written summary of what the team plans to accomplish in the next sprint.

2. *The sprint backlog:* The list of stories and other product backlog items the team has agreed to work on in the upcoming sprint.

Where does Sprint Planning take place?

A good location for sprint planning is the team room so that you have access to all the information about your product backlog and you can reference and update any information radiators you may use. If your team is distributed, sprint planning represents a good opportunity to gather everyone together so that your planning discussions can be more effective and to reinforce the person-to-person connections of the team.

There are *three main* things you should think about when you and your team members are choosing the next sprint backlog items to complete.

- **What:** This is the objective or goal you want to achieve during the sprint. The items you select from the project backlog should all tie back to the purpose of the upcoming sprint. Most of the time, the sprint goal ties back to an overarching goal, such as an OKR.

- **How:** This is the work that needs to be done to complete a backlog item, including the specific strategies that your team will use. Team members work with Scrum masters and product owners to identify if there is a specific way sprint backlog items need to be completed. This is all included in sprint planning. That way, when the sprint is in progress, the entire Scrum team knows exactly what they need to do and how to do it.

- **Who:** This is the group of people who are working on specific backlog items. The "who" is a key component of sprint planning sessions, because it prevents the overlap of work and ensures that the work being put into the sprint backlog doesn't exceed your team capacity.

  Highlighting who's responsible for each task also gives key stakeholders a clear point   of contact for each item in the sprint backlog.

Estimation is the process for estimating the effort required to complete a prioritized task in the product backlog. This effort is usually measured with respect to the time it will take to complete that task, which, in turn, leads to accurate sprint planning.

The Basic Project Estimation Process
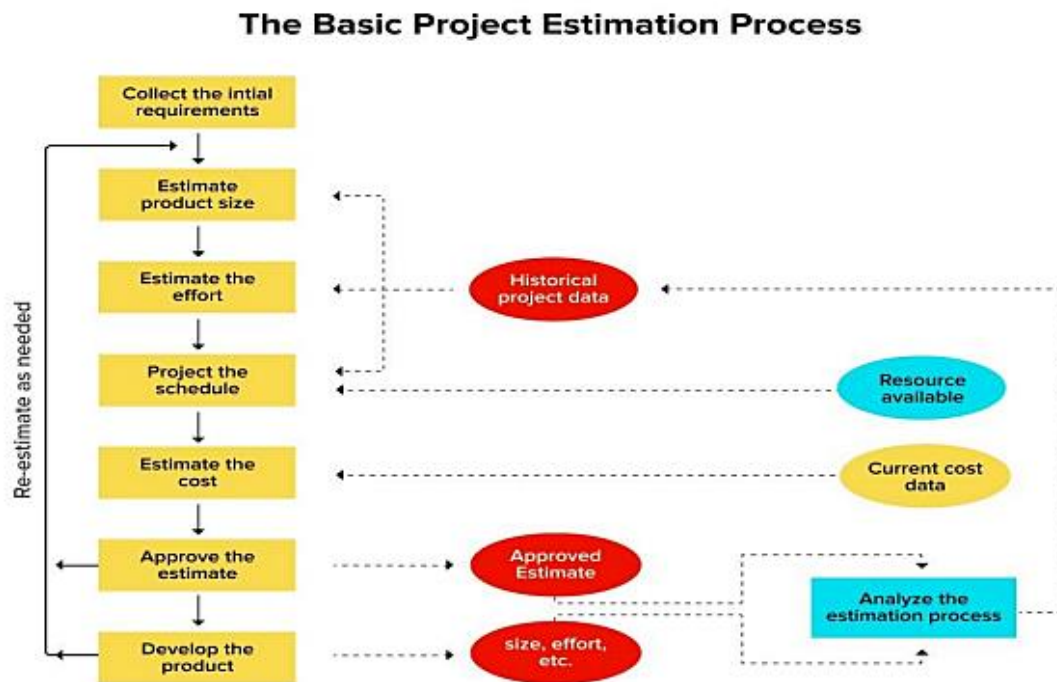
How To Create Accurate Sprint Estimates?

How do you know how much your team can get done in a sprint? You must know how long each task will take, which is also one of the trickiest parts of sprint planning.

Here's how to coordinate your team and make accurate sprint estimations:

Create Estimates as a Team:

In agile planning, the team members who will be doing the work collectively come up with the estimates. Everyone involved participates in the process so you can account for various perspectives and tap into diverse experiences.

Pick the Right Tools:

Whiteboards and sticky notes are no longer enough to support distributed teams and ensure that everyone participates equally. Collaboration software can help you coordinate the estimation process and gain insights into every aspect of the project.

Establish User Stories:

The foundation of most sprint estimation techniques is user stories, which describe the features from the users' perspective. There are many user story estimation techniques, and it's important to choose one that works well for your team and the nature of the project.

<u>Prioritize User Stories:</u>
Consider the value, cost, and risks associated with each feature. Under most circumstances, you'd try to maximize the value you gain with the lowest cost. But you may also want to tackle riskier tasks early on in the project to eliminate uncertainties or front-load features that will allow the team to gain knowledge to inform subsequent sprints.

<u>Convert Story Points into Hours:</u>

You can use story points to determine how long it'd take to complete a task, but you must first define the time it takes for your team to complete one story point worth of work. Start by calculating your team's pace, which uses historical data to correlate story points to time spent on a work item.

<u>Break down Big Tasks:</u>

If a task takes more than 8 hours, break it down into smaller apieces to help increase the accuracy of your estimation. You may have to move some parts to the backlog and tackle them in another sprint.

<u>Address Backlog Items:</u>

Review your product backlog to align priorities, maximize the value you get out of the sprint, and ensure that you have the right resources to get the work done. Estimating new backlog items can also help team members develop a shared understanding of the requirements.

<u>Revisit and Update Release Plan:</u>
Update your release plan after each sprint to incorporate learnings from the previous iterations, reassess priorities, address new risks, and adjust any change in pace to inform subsequent sprint estimates.

How to Estimate Sprint Velocity?

• Sprint velocity is the number of story points that can be completed during a sprint by a specific team.

• Unfortunately, there is not a concrete way to determine sprint velocity until the first sprint has been finished.

• However, the future velocity can be predicted by analysing the team's historical data (i.e., keeping track of how many story points were completed in a previous sprint).

• The total number of completed story points and measuring the team's actual velocity during a sprint can be used as a reasonable data count.

• This will help to determine how many sprint cycles will be required to complete the project

6.2 Sprint Delivery Schedule:

A Sprint schedule is a document that outlines sprint planning from end to end. It's one of the first steps in the agile sprint planning process—and something that requires adequate research, planning, and communication.

What to include in a Sprint Schedule:

1.Start and End Date:

> • In order to meet your deadlines, you have to provide specific start and end dates for projects and individual sprints. In general, it's a good idea to communicate with your team before sprint planning to make sure the dates are viable.
> • To illustrate, you may have several team members going out of town or working on other projects throughout the month. By researching availability ahead of time, it's easier to make sure the schedule works for everyone.

2. Sprint Goals:

> • Every software project and sprint need a clear and concise goals to be effective.
> • There are typically large-scale sprint goals, which may include tasks like building a website or mobile application. For such projects, there is usually one large goal and several underlying sprints with individual goals. If a project involves creating a chatbot for banking, a sprint goal might be to build as savings account system or a loan management system.
> • Given these points, you need to plan ahead when putting sprints together to make sure each one supports the ultimate task at hand.

Stories:

> • Within each backlog is a list of individual requests or user stories. For example, you may have requests for items like website buttons and user authentication features. Most product
> • owners go into sprint planning sessions with a list of outstanding user stories and narrow down the list together as a group.

Story Descriptions:

> • The final schedule should contain a list of specific tasks and instructions. The schedule should also clearly assign team members to individual tasks so that everyone is crystal clear about the items they own.
> • Under each task, you should provide brief descriptions outlining the main overarching story goal. Effective schedulers always strive to make it easy and intuitive for developers to follow along and complete objectives.

Why is Sprint Scheduling Important?

- Keep Projects on Track

- Maximize Efficiency

- Do Your Due Diligence

- Respect Other People's Time

How to make a Sprint Schedule?

- Check Your Roadmap

- Review Your Master Backlog

- Determine Your Resources

- Establish a Time Frame

- Propose a Schedule

- Finalize the Schedule

6.3 Reports from JIRA:

Stay on track of sprint goals and improve retrospectives with data scrum teams can put to use sprint over sprint.

Sprint Report: Determine over commitment and excessive scope creep and understand completed work in each sprint.

Burndown chart: Track progress towards sprint goals to manage progress and respond accordingly.

Release burndown: Track and monitor the projected release date for versions and take action if work is falling behind projected schedule.

Velocity chart: Track work from sprint to sprint to helps teams determine the velocity and better estimate the work a team realistically achieve in future sprints.

Sprint-1:

Sprint burndown   BETA ? ∨

20 points done, 0 points to go   ✓ On track

● Remaining work   ● Guideline

Sprint-2:

Sprint burndown   BETA ? ∨

20 points done, 0 points to go   ✓ On track

● Remaining work   ● Guideline

Sprint-3:

Sprint burndown   BETA ? ∨
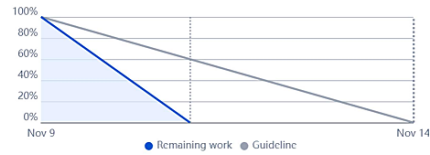
20 points done, 0 points to go   ✓ On track

● Remaining work   ● Guideline

Sprint-4:

Sprint burndown   BETA ? ∨

20 points done, 0 points to go   ✓ On track

● Remaining work   ● Guideline

Sprints

> ⚡ ABDFBI-19 Registration

> ⚡ ABDFBI-20 Login And Dashboard

> ⚡ ABDFBI-21 Account Creation and Details

> ⚡ ABDFBI-22 Existing User Support

> ⚡ ABDFBI-23 Loan Account Actions

> ⚡ ABDFBI-24 Net Banking Actions

> ⚡ ABDFBI-25 General Actions

> ⚡ ABDFBI-26 Admin Actions

7. CODING AND SOLUTIONING:

Feature 1:

Integrating Flask:

a. Flask is a web framework, it's a Python module that lets you develop web applications easily. It's has a small and easy-to-extend core: it's a microframework that doesn't include an ORM (Object Relational Manager) or such features.

b. It is a WSGI web app framework.

c. The Web Server Gateway Interface (Web Server Gateway Interface, WSGI) has been used as a standard for Python web application development. WSGI is the specification of a common interface between web servers and web applications.

Code:

```
1.  from flask import Flask, render_template
2.  app = Flask(__name__)
3.  @app.route("/")
4.  def bot():
5.      return render_template('Chatbot.html')
6.  if __name__ == '__main__':
7.      app.run()
8.
```
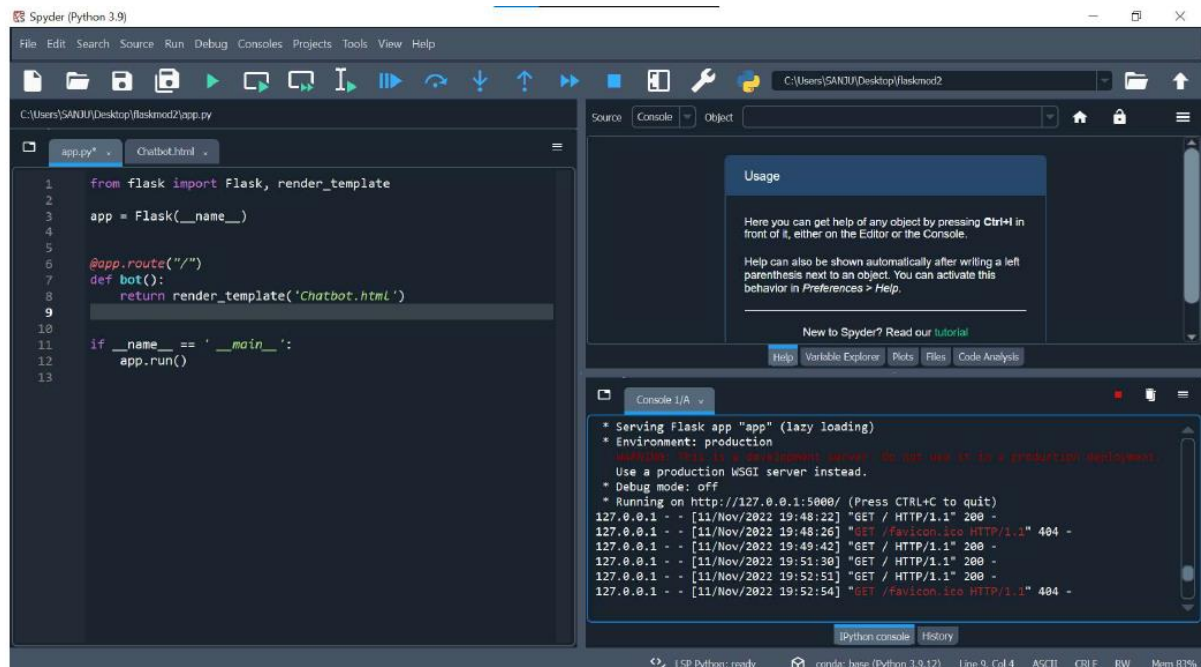
Chatbot.html file is linked through the code where this template is kept under a separate folder called as templates.

Chatbot.html code:

```html
1.  <html>
2.  <head>
3.  <meta charset="UTF-8">
4.  <title>Output</title>
5.  <link rel="stylesheet"
6.  href="https://maxcdn.bootstrapcdn.com/bootstrap/3.4.1/css/bootstrap.min.css". >
7.  <style>
8.  body
9.  {
10. background-image:url("https://i.postimg.cc/d1tLFVCd/chat.png");
11. background-size:cover;
12. }
13. </style>
14. </head>
15. <body>
16. <script>
17. window.watsonAssistantChatOptions = {
18. integrationID: "96c60b0a-3f34-427c-b477-ba9f879046f7", // The ID of this
19. integration.
20. region: "us-south", // The region your integration is hosted in.
21. serviceInstanceID: "fb9219fe-23de-484e-9f82-4b2ab6dcdc88", // The ID of your
22. service instance.
23. onLoad: function(instance) { instance.render(); }
24. };
25. setTimeout(function(){
26. const t=document.createElement('script');
27. t.src="https://web-chat.global.assistant.watson.appdomain.cloud/versions/" +
28. (window.watsonAssistantChatOptions.clientVersion || 'latest') +
    "/WatsonAssistantChatEntry.js";
29. document.head.appendChild(t);
30. });
31. </script>
32. </body>
33. </html>
```
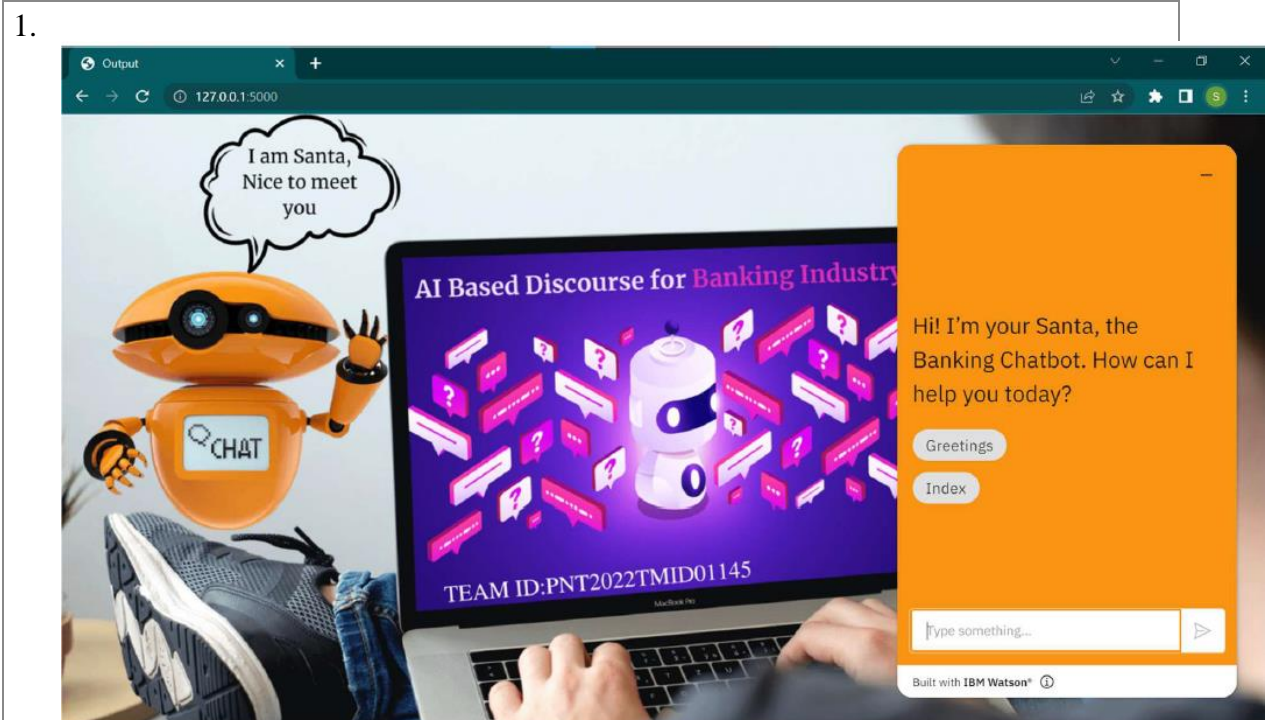
1.

Feature 2:

Integrating with Slack:

a. Slack is a messaging app for business that connects people to the information that they need. By bringing people together to work as one unified team, Slack transforms the way that organizations communicate.

b. Slack is an instant messaging program designed by Slack Technologies and owned by Salesforce.
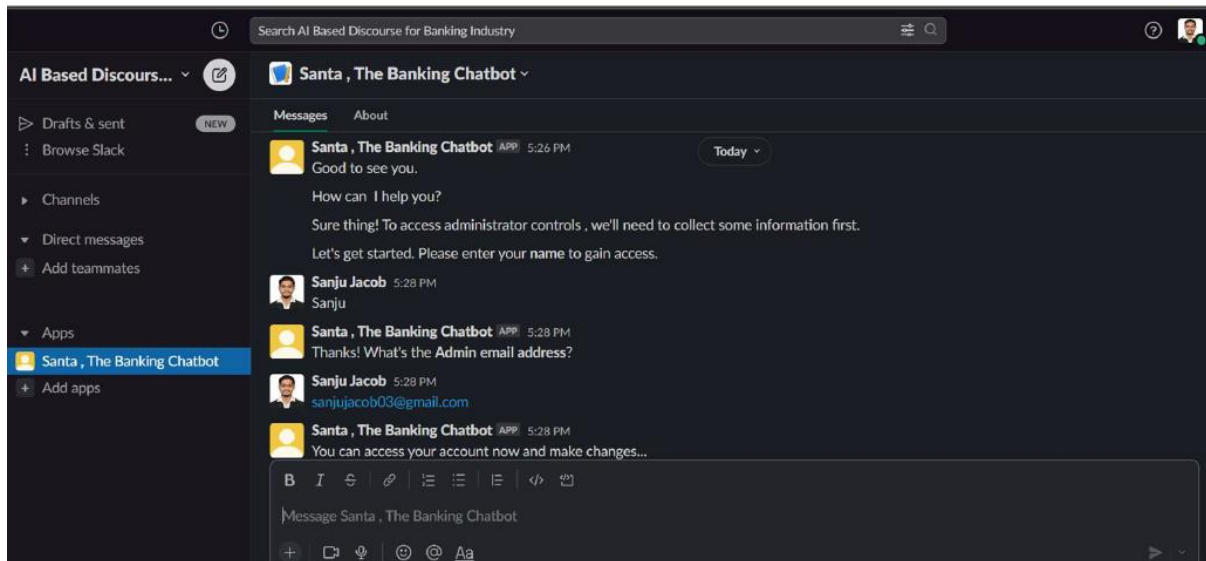
Feature 3:

Integrating Facebook:

a. Facebook is an online social media and social networking service owned by American company Meta Platforms. Founded in 2004 by Mark Zuckerberg.

a. Facebook can be accessed from devices with Internet connectivity, such as personal computers, tablets and smartphones. After registering, users can create a profile revealing information about themselves
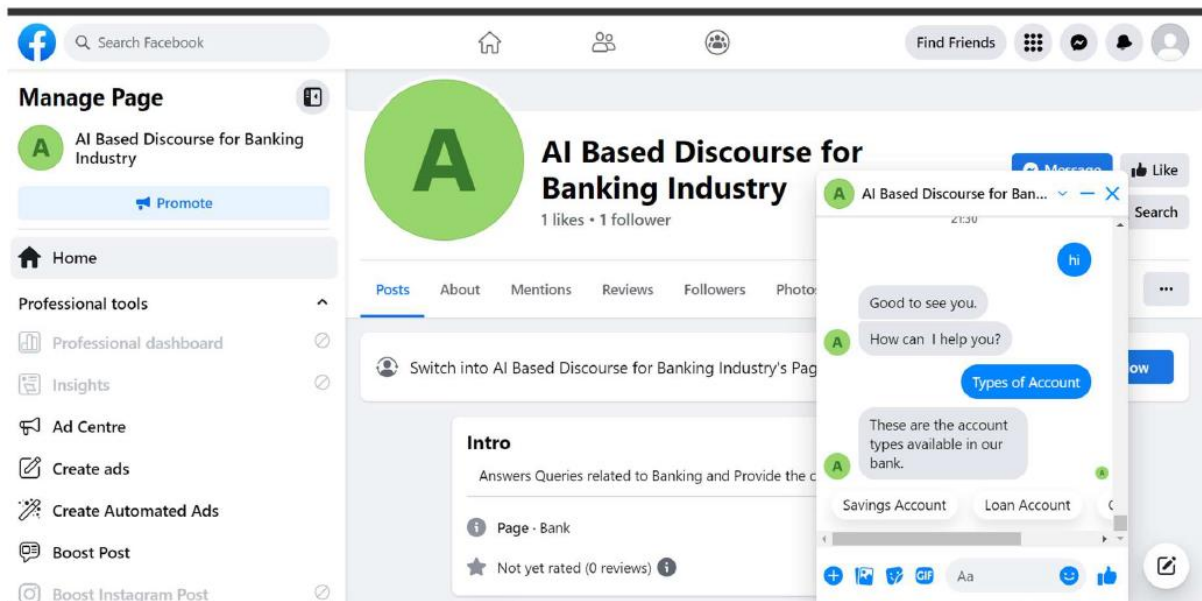
Overall Architecture after deployment:

8. TESTING

Software testing is the process of evaluating and verifying that a software product or application does what it is supposed to do. The benefits of testing include preventing bugs, reducing development costs and improving performance.

8.1 Test Cases:
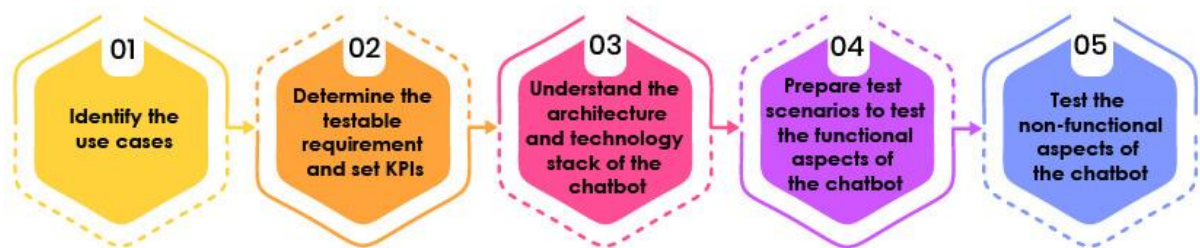
1. Identify the use cases:

To start with the testing process, first identify the use cases for the chatbot. It is essential to list all the questions and responses to cover every scenario and prioritize them accordingly.

2. Determine the testable requirement and set KPIs:

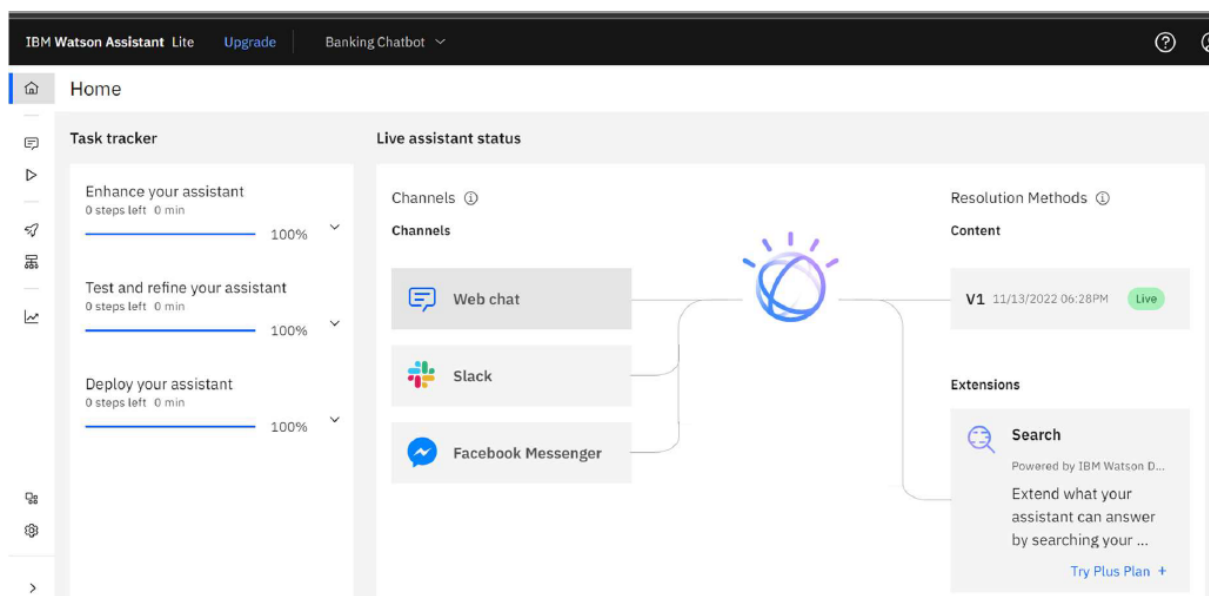Once the use cases are identified, it is essential to see what needs to test based on the identified use cases. Also, set the KPIs such as self-service rate, performance rate, usage rate per login, escalation rate, etc.

3. Understand the architecture and technology stack of the chatbot:

Once all the use cases are identified and KPIs are set, it is essential to understand the underlying architecture and technology used by the chatbot for each use case.

4. Prepare test scenarios to test the functional aspects of the chatbot:

Prepare the test scenarios such that it covers functional aspects of the AI chatbot testing. Test scenarios should cover the conversation and voice testing. It is necessary to prepare scenarios to see how the chatbot responds to the same inputs and handles errors.

5. Test the non-functional aspects of the chatbot:

From the non-functional viewpoint, the various factors to be tested include performance, load handling capacity, security, and accessibility of the chatbot

8.2. User Acceptance Testing:

User satisfaction matters a lot for businesses today. Therefore, to ensure the bots deliver a seamless CX, businesses must leverage user acceptance testing. By including end-users in the testing process, businesses get real-time feedback from users, which helps to improve product quality and CX.

UAT DESIGN

In UAT design phase, test engineers are preparing UAT testcases as per the business requirements. AT

test coverage should be with Alpha and beta testing . After having complete idea about business

requirements and have a discussion with BA or Product Owner one can proceed with UAT Test case design/mapping to UAT test suite.

Approach: Alpha Testing and Beta Testing

• UAT Test scenarios & Testcases prepared based on business needs in both functional and non-

functional aspects

• UAT Testcases can be sub set of existing testcases and maintained as a separate UAT Test suite

• UAT Test scenarios & cases once designed should be reviewed by BA or Product Owner

• UAT Testcases target is customer environment based in-terms of Test data and Servers

10. ADVANTAGES AND DISADVANTAGES:

ADVANTAGES (PROS):

1. Speed and availability. Bots have the benefit of handling a greater volume of calls at higher speed than the hours and capacity that human operators keep.

2. Consistency without the fatigue of a human operator.

3. Cheaper than a call centre, with no time spent "on hold "

4. Chats can be saved to help provide an audit trail or customer history if logged in.

5. Analytics can help to capture trends or errors on your site or service faster.

6. Your customers won't have to take notes or remember what they were told if you can email them the transcript of the interaction.

7. Having a chat window open may offer your customer better multi-tasking opportunities than being on the phone.

8. It can be more discrete for customers, they would not be overheard in their home or office, and they can ask questions anonymously if contact details are not required.

9. Quick, accurate responses can help with customer satisfaction and therefore loyalty.

10. It is easy to provide more detailed information in a link, rather than follow-up or explain in great detail on a call.

DISADVANTAGES(CONS):

1. Use of bots usually comes at the cost of authenticity empathy which a human can provide.

2. Often bots will feel more scripted which can cause frustration.

3. For bots and human operators, it is important not to have answers that stray into financial advice as this is regulated.

4. The information has to be up-to-date at all times.

5. Depending on the technology it may have issues with dealing with mis-spelled or mis-typed words.

6. It requires users to be a little more tech-savvy than a phone call or in person interaction.

7. It requires the internet and as such may not be a solution for all of your customers.

8. The expectations on response time is higher than with call-centre's, seamless handovers to human operators is a must with expectations set.

9. If you are operating with secure account information you need to have a way of resolving identity.

10. With secure account information it opens up the potential for another avenue of phishing attacks.

11.

11. CONCLUSION

- As we can see, chatbots and other types of AI assistants are of great use in any industry that has to provide high-quality customer support. One such industry is the finance or banking area, and it is rapidly integrating these technologies into its workflow. Banking is all about money and reputation, and AI chatbots offer numerous benefits for both. Chatbots are being adopted in the banking sector at a very fast rate.
- They are not only being used for answering customers questions but also for providing a wide range of services which include bill payment, fund transfer, view recent transactions and much more. Chatbots are also getting smarter due to integration of natural language processing and machine learning. By helping customers round the clock, they help banking staff focus on other important tasks. Therefore, we can say that chatbots have become an essential part of the banking system.

12. FUTURE SCOPE

The share of banks that use AI solutions and chatbots in particular is constantly rising. As another factor, the use of smartphones and other smart devices is also a rapidly growing trend. These two driving forces determine the near future of artificial intelligence assistants in the banking industry. More and more banks tend to integrate chatbots into their mobile apps. This is a convenient way to stay in touch with their clients and, at the same time, reduce the involvement of human personnel. According to estimations calculated by Juniper Research, in 2023, chatbot interactions will save 862 million hours for banks, which equals to $7.3 billion cost savings worldwide.

KEY MARKET STATISTICS FOR AI IN BANKING

$7.3 Billion – Total banking chatbots cost savings for businesses in 2023

3,150% – Growth in successful banking chatbot interactions 2019-2023

862 Million Hours – Total time saved for banks by chatbot interactions in 2023

79% – Share of successful chatbot interactions via mobile banking apps in 2023

The quality of chatbots will definitely improve over the next few years. They will become more "human," and will learn to interpret requests much better. As a further development, chatbots will predict human behaviour more accurately and use this information for self-learning.

In the nearest years, we will definitely see the rise of voice assistants in all fields, including the banking sector. For example, they will greatly speed up the process of communication with clients. The supported functionality will also grow and will result in more use cases for chatbots.

Security concerns regarding the use of chatbots in the banking industry will persist in the future. The need for well-protected and reliable AI solutions will become a major driving force of digital technologies development. The reason is simple: people will agree to share their private information and access to their credit cards only with the most protected and trustworthy software solutions, including chatbots.

Banking systems have long been a popular use case for chatbots, but the technology has only recently become sophisticated enough to make them truly useful. Initially, chatbots were simple FAQ engines that could only answer questions already available in their database. With the rise of ML and NLP (Natural Language Processing), bots can have more meaningful conversations by interpreting questions that are not within the database. However, when it comes to responding – the responses are usually templated.

In the future, we can expect more non-templated responses from bots. We are fast advancing to a time when it may be difficult for a customer to guess whether they just chatted with a human or a machine.

13. RESULTS:
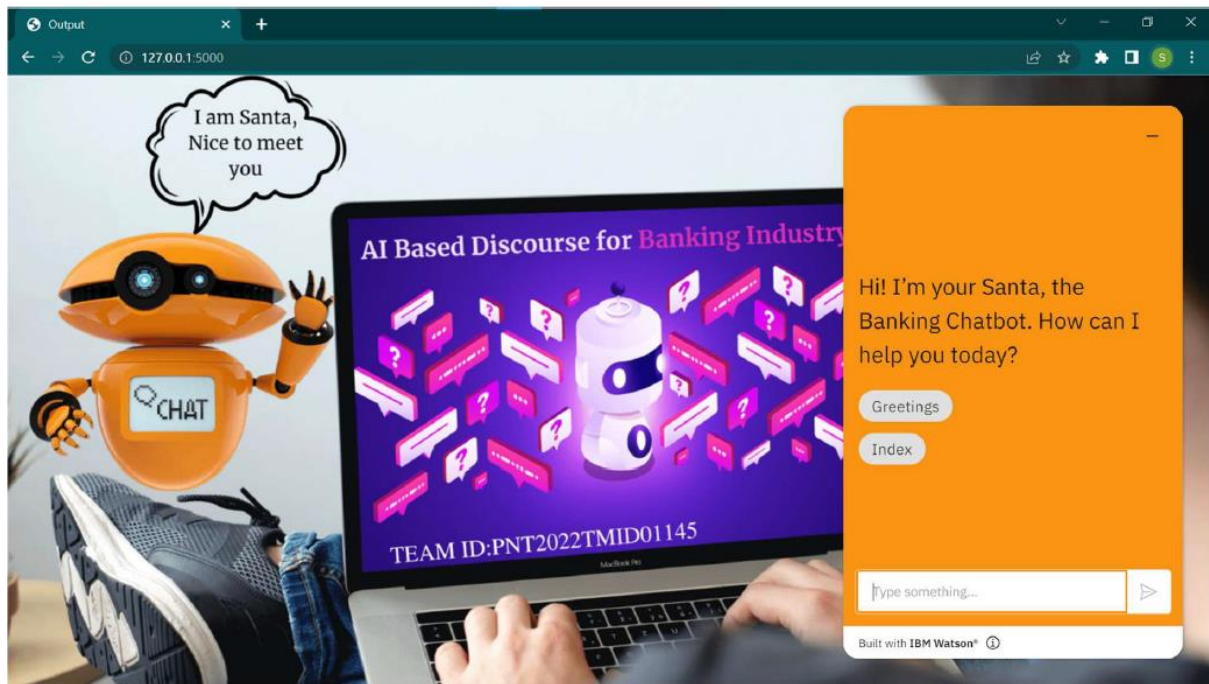
Performance Testing:

Performance testing is a form of software testing that focuses on how a system running the system performs under a particular load. This is not about finding software bugs or defects. Different performance testing types measures according to benchmarks and standards. Performance Testing gives developers the diagnostic information they need to eliminate bottlenecks.

13. APPENDIX

Source Code:

Flask

```
1.  from flask import Flask, render_template
2.  app = Flask(__name__)
3.  @app.route("/")
4.  def bot():
5.    return render_template('Chatbot.html')
6.  if __name__ == '__main__':
7.   app.run()
```

HTML Code:

```html
1.  <html>
2.  <head>
3.  <meta charset="UTF-8">
4.  <title>Output</title>
5.  <link rel="stylesheet"
6.  href="https://maxcdn.bootstrapcdn.com/bootstrap/3.4.1/css/bootstrap.min.css". >
7.  <style>
8.  body
9.  {
10. background-image:url("https://i.postimg.cc/d1tLFVCd/chat.png");
11. background-size:cover;
12. }
13. </style>
14. </head>
15. <body>
16. <script>
17. window.watsonAssistantChatOptions = {
18. integrationID: "96c60b0a-3f34-427c-b477-ba9f879046f7", // The ID of this
19. integration.
20. region: "us-south", // The region your integration is hosted in.
21. serviceInstanceID: "fb9219fe-23de-484e-9f82-4b2ab6dcdc88", // The ID of your
22. service instance.
23. onLoad: function(instance) { instance.render(); }
24. };
25. setTimeout(function(){
26. const t=document.createElement('script');
27. t.src="https://web-chat.global.assistant.watson.appdomain.cloud/versions/" +
28. (window.watsonAssistantChatOptions.clientVersion || 'latest') +
    "/WatsonAssistantChatEntry.js";
29. document.head.appendChild(t);
30. });
31. </script>
32. </body>
33. </html>
34.
```

GitHub link:

https://github.com/IBM-EPBL/IBM-Project-197-1658220994

YouTube Demo link:

https://www.youtube.com/watch?v=C0CyZ2xI2h4

TEAM MEMBERS:

SAKTHIVEL P.
SANJU K. JACOB
SARABESH KANNAA T.M.
VIRUTHIK KUMARESAN S.