**Agentic AI Hackathon**

|  |  |
| --- | --- |
| **Team Section** | |
| Name of team | Mr Anderson |
| Team Member 1 | Marwan Attar |
| Team Member 2 | Agent Smith |
| Team Member 3 | Agent Jones |
| Team Member 4 | Agent Brown |
|  | |

|  |  |  |
| --- | --- | --- |
| **Use Case Section** | | |
| Name of use case | | Sales Onboarding |
| Industry of use case | | Applicable to any industry |
| Description of use case | | New sales representatives joining a company (or existing reps joining a new team) might, at first, and especially at the beginning of Q1, find it overwhelming to trawl through the mountain of information required to generate viable and actionable sales plans during the account planning phase.  Common questions include:   * What are my account(s) this year? * Which representatives were aligned to my account(s) last financial year? * What were the notable actions taken against said account(s) last financial year? (for instance, ELA, purchase of net new IBM Cloud Pak Licenses, change of leadership, mergers, ongoing RFIs and so on so forth) * Short and long-term strategies pertaining to said account(s) * Existing installs/footprints * Unique insights & lessons learnt * Etc.   As you can see, this information is typically distributed across different data sources. Taking an IBM lens, existing installs can be found in FastPass/SQO, whereas notable actions, stakeholders and strategies can be found in a combination of Box/OneDrive, Quip and Salesforce. Furthermore, while Quip plans are typically one pagers (ie, one document), this is not the case with Box/One Drive, where information pertaining to an account is typically distributed against a number of documents.  To top it off, a new joiner would typically request access to those systems, which can take a couple of days to approve, further increasing the delay. As a result, new joiners would typically set some time with the respective representatives for, depending on circumstances, a handover or an enablement, which typically takes hours.  In an attempt to have those questions answered with minimal latency, and to reduce load on existing reps during such a critical phase, one might propose the deployment of an autonomous system capable of translating natural language into a set of questions/queries (where each question is a function of the target data source containing the information desired) performed against the target data source(s).  This will ensure new joiners can hit the ground running in the shortest time possible whilst reducing the cognitive load on existing reps. |
| Challenges/hopes of each persona | | |
| • | Persona A  New Representatives | New Representatives  New reps might find the overall onboarding process a little overwhelming initially, namely due to the distributed nature of modern applications. An elevated user (typically front-line managers) would first have to grant access to said applications. Next, and in the case of Box, information pertaining to the account(s) might be found in a folder with many contributors and files within. Initially, while new joiners need not concern themselves with the finer details at such an early stage, summarising the files within the folder is a non-trivial lengthy task.  Having key information distilled to the end user in the shortest time possible is key. This also has the side effect of minimising “busy waiting” times (busy waiting on access to a service, or busy waiting on an existing rep’s availability for either an enablement or a handover session) |
| • | Persona B  Existing Representatives | Existing Representatives  *(This solution is not suggesting the elimination of handovers and/or enablement sessions)*  On the contrary, handovers and enablement sessions are of utmost importance and are here to stay. That said, there exists scope for perhaps decreasing the time allocated to said handovers assuming our new joiner (Persona A) already has access to the required information and has, through the proposed tooling, managed to make sense of the information.  This will allow our persona to allocate more time on other, perhaps more pressing immediate tasks. |
| As-is process (without Agentic AI) | | This was described above but here are the (typical) sequence of steps a new sales representative encounters in the first couple of weeks:   1. Asking (typically the assigned FLM) what system/sources should the sales representative require access to? 2. Request access to said systems/data sources 3. Wait X days/hours for access 4. (Optional): While waiting for access, set some time with existing reps for handover/enablement sessions 5. Access granted to required systems 6. Read, digest and summarise information made available to the new joiner (as a result of access granted to systems) 7. Set some time with existing reps to confirm new joiner’s findings from the above step 8. Hit the ground running!   Our new system, described below, can effectively eliminate Steps 1 through to 5 above. Our Agentic AI would, with minimal latency, surface the answer to the user’s query(ies) to be  The confirmation step is crucial and should not be omitted. There is always a risk with LLM’s hallucinating and present outdated or inaccurate information. While guardrails can be implemented to minimise such occurrences, they can’t be eliminated. |
|  | | |

|  |  |  |
| --- | --- | --- |
| **Solution Section** | | |
| To-be process (with Agentic AI) | | Mr Anderson, our virtual assistant, shall function as an AI-powered virtual assistant that shall act as a single, intelligent, interface between the various data sources for new joiners to the company. Behind the scenes, it leverages the agentic AI framework to query orchestration and forward (transformed) queries to legacy tools to complete requests.  It is also employee “aware”, in the sense it is able to able to schedule Calendar meetings on behalf of the new joiner, through its integration with Microsoft Graph.  Steps 1 through to 5 described above can be eliminated. Step 6 can be shortened, while step 7 can be automated.  Truly a win-win situation for all!  **Key Features & Functionalities**   * **Self-service** experience * Eliminates direct access to multiple systems or * portals by funnelling requests to **a single point**. Systems include databases and object storage (ie, Box). * Advanced **Search** and **Summarisation** capabilities, able to summarise lengthy (or many short documents) into one single concise document * AI agent securely retrieves **employee-specific** data from legacy systems and databases. * Ensures role-based access control (**RBAC**) for data privacy and compliance. |
| Business value statement | | |
| • | Persona A | As stated above, Persona A, our new joiner is now able to contribute and hit the ground running in a shorter amount of time as a result of:   * Not needing to wait for access to multiple systems * Advanced summarisation capabilities of Mr Anderson. The many documents in, say, Box can be summarised into one concise “all you need to know” Box note.   All that is required is access to Mr Anderson. |
| • | Persona B | Persona B, our existing representative is now able to allocate only a fraction of the time he/she would have otherwise allocated for enablement/handover in the past.  In both cases, that is, Persona A and B, many hours can be saved, increasing operational efficiency as a result. |
| Architecture | | System Architecture can be found below. We use the open-source CrewAI framework to orchestrate the workflow and the sharing of information from one agent/task pair to the other.  In chronological order, the flow is as such:   * The new representative wishes to understand which representative to reach out to seek information in relation to onboarding to a new account. * The query, via the appropriate agent, is dispatched to the appropriate data source (user/account DB in this case), to fetch the identity of the existing rep. The original query is converted to a SQL query using the relevant model available on the WatsonX platform. * Working (flagged as non-sensitive) documents, belonging to the existing rep is fetched from Object Storage (eg, Box) pertaining to said account. * This data (byte stream) is sent to the summariser agent, and a summary request is dispatched to the relevant model available on the WatsonX platform. * The summary is returned to the user, and a meeting is scheduled with the appropriate rep to confirm the user’s understanding of the account. * Optionally and as indicated by the dashed arrows, user analytics can be obtained by asynchronously pushing request/response pairs to a message/event broker (such as MQ or Event Streams or even a combination of both if exactly once delivery is required) for storage into a FAQ database or similar should a “complete” solution be desired. (CrewAI have callbacks on a per task basis which can be used for this purpose   At the time of writing, I already have made a custom demo for this. The user/account database and cloud object storage were simulated by means of Docker containers, PostgreSQL and an Nginx server respectively. |
|  | | |

NOTE: By the time the “Review” phase of this Hackathon is begun, I would’ve left IBM as I have already resigned. I’ve been out of the AI game for a number of years and so participating in this hackathon proved a good learning experience for me. Now, in the (extremely unlikely) event I win one of the prizes, please disregard and send the BluePoints to the “next in line” participant(s).