In recent years, the livestreaming industry has been gaining tremendous popularity. Apart from individual streamers, many businesses are also leveraging live streams to boost their revenue. DAE as a fundraising organization, we too aspire to benefit from this trend. However, due to budget constraints, we couldn't afford a human host. As an alternative, we have acquired an older model of a robot for our live-stream events.

Our plan is to have the robot interact with the audience, showcasing the cutting-edge research capabilities of our university to achieve our fundraising goals. However, due to the outdated model of the robot, it can only perform pre-programmed actions and basic text-to-speech. Moreover, there is a risk of the robot overheating and potentially exploding if subjected to excessive operations over a short period. Fortunately, the robot supports modern language programming and can be connected to the internet, enabling us to operate it remotely.

In addition to the robot, our company has the following internal equipment systems:

1. Our own API server. The server is also an older model and cannot perform highly complex calculations.
2. A photography terminal, which may be an older model smartphone or laptop.
3. The database we use in our daily operations boasts extensive user data and a highly intricate data model that we take pride in.

With only these devices, we would likely be unable to showcase any high-tech research capabilities. Therefore, the university has generously provided us with a free cloud server. This cloud server is exceptionally powerful, encompassing all the outward-facing functionalities offered by current platforms such as Google and Amazon. Essentially, any commercially available feature can be found within this cloud server.

1. Considering the information provided earlier, develop a solution to facilitate our live-streaming fundraising initiative. Utilize a sequence diagram to illustrate the proposed architecture.
2. Create an Entity-Relationship (ER) diagram reflecting this architecture
3. Present high-level code in C# to enable the robot to respond effectively to users' streaming chat input.
4. Provide sql snippets to support your code in (3.) and based on your ER diagram