# Design

This section describes the design of the system, including functional requirements, frontend design, and class diagrams.

## Software Requirements

### Introduction

Based on conversations with the three beta testers, as well as considering time and complexity constraints, the following functional requirements were outlined. These focus on the two key ontologies that the users wanted to ask questions about – Person and Country.

### Functional Requirements

#### 1.0.0 User Interaction

* 1.1.0 The application should allow the user to interact with the chat bot
* 1.1.1 The user should be able to access the chatbot in a browser
* 1.1.2 The user should have a text box to type their query
* 1.1.3 The user should clearly see the response of the chatbot in the webpage
* 1.1.4 The user should be able to clearly see their conversation with the chatbot in the webpage

#### 2.0.0 Queries

* 2.1.0 The chatbot can answer basic questions about people:
  + 2.1.1 The application should take a user query about a person - ‘who is X’ - and respond with a description of that person.
  + 2.1.2 The application should take a user query about the birthdate of a person – ‘when was X born’ and return the birthday of the given person.
  + 2.1.3 The application should take a user query about the age of a person - ‘how old is X’ - and return the age of the given person.
  + 2.1.4 The application should take a user query about the birth place of a person – ‘where was X born’ - and return the birth place of the given person.
  + 2.1.5 The application should take a user query about the death date of a person – ‘when did X die’ and return the birth place of the given person.
  + 2.1.6 The application should take a user query about what a person is known for – ‘what is X known for’ and return a description of what the given person is known for.
  + 2.1.7 The application should take a user query about what a person looks like – ‘photo of X’ or ‘what does X look like’ - and return a photo of the person
  + 2.1.8 The application should take a user query about linking to the Wikipedia page of a person, and return a link to that page.
* 2.2.0 The chatbot can answer questions about countries:
  + 2.2.1 The application should take a query about the population of a country, and return the population of that given country.
  + 2.2.2 The application should take a query about the capital of a country, and return the capital of the country.
* 2.2.3 The application should take a query about the description of a country, and return the description of that given country.
* 2.2.4 The application should take a query about the flag of a country, and return the flag image of that given country.
* 2.2.5 The application should take a query about where a country is on a map, and return a google maps location of that country.

#### 3.0.0 Advanced Queries

* + 1. The chatbot can perform advanced query searches and comparisons:
* 3.1.0 Advanced People queries
  + 3.1.1 The user can ask for a list of people born in a given year, e.g. ‘who was born in 1995’ and the chatbot returns a list of notable people born in that year
  + 3.1.2 The user can ask for a list of people born in a given place, e.g. ‘who was born in London’ and the chatbot returns a list of notable people born in that place
  + 3.1.3 The user can ask for winners of a given prize, e.g. ‘who won the Nobel Peace Prize [in 2009]’ and the chatbot returns a list of winners of that prize, or the winner of the prize in a given year.
* 3.2.0 Advanced Country queries
  + 3.2.1 The user can ask the chatbot for a list of countries with a given official language, e.g. ‘which countries speak Italian’, and the chatbot will return a list of countries which identify that language as their official language.

#### 4.0 Conversation

* 4.1 The user should greet the chat bot and be returned with a similar greeting – e.g. Hello.
* 4.2 The user should be able to ask for example queries and the chat bot returns a number of working example queries
* 4.3 The user should be able to ask for help using the chatbot and be returned with a statement about how to use the chat bot.

### Performance Requirements

#### 5.1 Performance

* 5.1.1 The web page should load fully in less than 5 seconds
* 5.1.2 The chat bot should respond to each query within 5 seconds

#### 5.2 Reliability

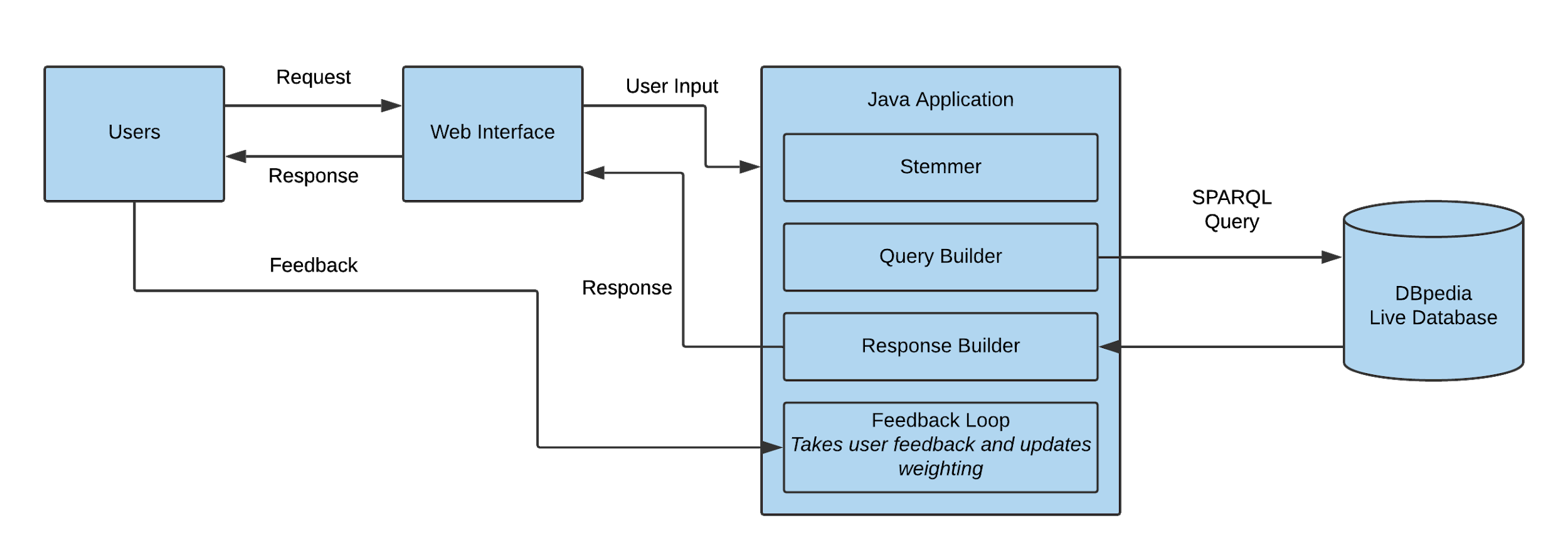
* 5.2.1 The application should function without failure
* 5.2.2 Any errors that do occur during normal operation should be logged, and the user should be clearly informed that an error has occurred.

## User Interface

* User interface design
* Flowchart

## Architecture

* System design



* Class Diagram