|  |  |
| --- | --- |
| City Chatbot Software Requirements Specification (SRS)  UMGC Fall 2020 | Benjamin Fetterman, Benjamin Murray, Hanim Danur, James Cornelius, Robert Lee  SWEN 670 |

Software Requirements Specification (SRS) Approvals

|  |  |  |
| --- | --- | --- |
| Name | Signature | Date |
| Approved by:  Dr. Mir Assadullah |  |  |
| Approved by:  "Stakeholder" |  |  |
| Approved by:  "Project Manager" | Robert Lee | 9/5/2020 |

Revision History

|  |  |  |  |
| --- | --- | --- | --- |
| Revision | Date | Name | Description of Change(s) |
| 1.0 | 9/5/2020 | Robert Lee | Document Initiated |
| 2.0 | 10/31/2020 | Robert Lee | Updates based on production version. |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

Table of Contents

[1 Introduction 5](#_Toc55320655)

[1.1 Purpose 5](#_Toc55320656)

[1.2 Scope 5](#_Toc55320657)

[1.3 Abbreviations, Acronyms and Definitions 5](#_Toc55320658)

[1.3.1 Acronyms 5](#_Toc55320659)

[1.3.2 Definitions 5](#_Toc55320660)

[1.4 References 6](#_Toc55320661)

[2 Overview 6](#_Toc55320662)

[2.1 Overall Description 6](#_Toc55320663)

[2.2 Functional Requirements 6](#_Toc55320664)

[2.3 Use Case Diagram 8](#_Toc55320665)

[2.4 Use Cases 9](#_Toc55320666)

[2.5 Chatbot Stimulus Response / Data Flow Diagram 13](#_Toc55320667)

[2.6.1 Chatbot – Greeting, CH-1 1](#_Toc55320668)

[2.6.2 Chatbot – Zoning/Permit Links, CH-2 1](#_Toc55320669)

[2.6.3 Chatbot – Application Links, CH-3 2](#_Toc55320670)

[2.6.4 Chatbot – Retrieve Regulations, CH-4 2](#_Toc55320671)

[2.6.5 Chatbot – Retrieve Standards, CH-5 3](#_Toc55320672)

[2.6.6 Chatbot – Retrieve Permits Application to Zoning Status, CH-6 4](#_Toc55320673)

[2.6.7 Chatbot – Chatbot GIS Map Layer Integration, CH-7 5](#_Toc55320674)

[2.6.9 Chatbot – Address Validation, CH-9 6](#_Toc55320676)

[2.6.10 Chatbot – Other Information, CH-10 7](#_Toc55320677)

[2.6.11 Chatbot – Help, CH-11 7](#_Toc55320678)

[2.6.12 Chatbot – Get Started, CH-12 8](#_Toc55320679)

[2.7 User Classes and Characteristics 1](#_Toc55320680)

[2.8 Assumptions and Dependencies 1](#_Toc55320681)

[2.8.1 In Scope 1](#_Toc55320682)

[2.8.2 Out of Scope 1](#_Toc55320683)

[3 Software Requirements 1](#_Toc55320684)

[3.1 Developmental, Software Requirements 1](#_Toc55320685)

[3.2 Operational, Software Requirements 2](#_Toc55320686)

[3.1 Major System Components 2](#_Toc55320687)

[4 External Interface Requirements 2](#_Toc55320688)

[4.1 User Interfaces 2](#_Toc55320689)

[4.2 Hardware Interfaces 4](#_Toc55320690)

[4.3 Software Interfaces 4](#_Toc55320691)

[4.4 Communications Interfaces 4](#_Toc55320692)

[5 Other Nonfunctional Requirements 4](#_Toc55320693)

[5.1 Performance 4](#_Toc55320694)

[5.2 Security 5](#_Toc55320695)

[5.3 Availability 5](#_Toc55320696)

[5.4 Technology 5](#_Toc55320697)

[Appendix A: Pending Admin Functionality 6](#_Toc55320698)

List of Figures

[Figure 1 Use Case Diagram of System 9](#_Toc55320600)

[Figure 2: Data Flow Diagram 14](#_Toc55320601)

[Figure 3 Diagram of Data Flow in System 1](#_Toc55320602)

[Figure 4 Chatbot Start Icon 3](#_Toc55320603)

[Figure 5 Chatbot Asks User for Input 3](#_Toc55320604)

[Figure 6 User Can Enter Input 4](#_Toc55320605)

1 Introduction

The City of Pasadena City Chatbot, here unto described as the system, application, or agent; provides zoning level and information-based support for customers of the city’s zoning and planning body. The application intends to be in use by city residence and provide information regarding regulations, permits, applications, and standards within the City of Pasadena zones. The application is an intermediary actor between an organization and its customers/users. The agent resides within IBM Cloud® infrastructure as a virtual service. This service processes natural language to fulfill costumer inquires in a familiar manner. The virtual chatbot agent delivers user specific solutions to zoning and planning needs, while migrating operational burden from human capital to an information framework open to extensibility.

1.1 Purpose

This document describes the system’s functional requirements and goals. It provides a scope of the system’s parameter based limitations and expectations, an exposition of it as a system, use cases, information management system, and basic hardware and software requirements.

1.2 Scope

The system provides the customers with City of Pasadena regulatory standards for permits, development, and applications. The agent will validate customer input, in the form of a city address, with a multilayer map. Converting the given address into latitude and longitude coordinates, narrowing down fulfillment to zone specific permit, application, standard, and/or regulation information when in need.

The system is a flexible and dynamic framework, such that the configuration to different regions requires a minimal level of effort.

For this **version** of the system, cities other than Pasadena are not in scope, information other than regulations and permits are not in scope. All development is to be open source or free based solution. Any application that is to be used by the Chatbot team that they did not develop must be open source as well. This version represents a proof of concept, proving a stable, flexible, and extensible architecture.

1.3 Abbreviations, Acronyms and Definitions

1.3.1 Acronyms

Artificial Intelligence (AI)

Application Programming Interface (API)

Hypertext Transfer Protocol (HTTP)

Software Requirements Specification (SRS)

Graphical User Interface (GUI)

1.3.2 Definitions

Chatbot: A application that recognize language and responds to user inputs and requests

DevOps: Software Development and Information Technology joined into a single process to provide continuous delivery

Scrum: Agile software development methodology which accentuates, a backlog list of requirements, cyclical development cycles, sprints, and tracking meetings.

1.4 References

| **Title** | **Reference Location** |
| --- | --- |
| ESRI. (n.d.). Make Maps. ArcGIS Online | <https://www.esri.com/en-us/arcgis/products/arcgis-online/capabilities/make-maps> |
| IBM. (2020, March 12). Watson Assistant API overview. IBM Cloud Docs. | <https://cloud.ibm.com/docs/assistant?topic=assistant-api-overview> |
| OpenLayers. (n.d.). Editable ArcGIS REST Feature Service. | <https://openlayers.org/en/latest/examples/vector-esri-edit.html?q=arcgis> |
| Municipality Permit ChatBot Software Requirements Specification (SRS) For UMGC City Application Version 1.2 (2020, July 23) | <https://onedrive.live.com/view.aspx?cid=d159c6583f3538af&page=view&resid=D159C6583F3538AF!286&parId=D159C6583F3538AF!276&authkey=!AA0CVC0ilkS0fOY&app=Word> |
| SonarQube, Documentation | <https://docs.sonarqube.org/latest/> |
| Java Spring.io Framework | <https://spring.io/projects/spring-framework> |
| IBM Cloud, Documentation | <https://cloud.ibm.com/docs> |
| GitHub | <https://docs.github.com/en/free-pro-team@latest/github> |
| MySQL, Documentation | <https://dev.mysql.com/doc/> |
| MapQuest GeoCoding, Documentation | <https://developer.mapquest.com/documentation/geocoding-api/> |

2 Overview

This section reports on a brief description of the major system components, and delves into a more detailed narration of the system in the form of functional requirements, use case diagram, use cases, Chatbot Stimulus Response / Data Flow Diagram

2.1 Overall Description

The major functional components of the system are the IBM Watson Chatbot agent, custom API java spring.io service layer linking the MapQuest’s map layer geocoding API validating address and zinging information. As well as a back-end MySQL database. For exceptions outside of requirements user fulfillment is a general help message with additional information and URL for the official City of Pasadena’s zoning commission website. The chatbot provides information in the form of details, and links for applications, permits, regulations, and development standards. Linking the map layer geocoding to chatbot provides geolocation validation and city zone level specific information, such as the applications, permits, regulations, and standards and related information per zone category. Chatbot help provides the user information and directions about the chatbot and its use. The other information to catch exceptions outside of requirements, is a link to frequently asked questions.

2.2 Functional Requirements

Functional requirements are descriptions of what the system shall do. They describe the intended behavior of the system.

| **ID** | **Name** | **Description** |
| --- | --- | --- |
| CH-1 | Chatbot | The system shall have a chatbot. |
| CH-2 | Chatbot – Greeting | The chatbot shall have a greeting. This greeting shall direct the input of the chatbot user. |
| CH-3 | Chatbot – Allowed Land Uses | The chatbot shall provide zoning and permit links and information. Such as:   * Definition * Land use category * Variance * Entitlement process where applicable for Conditional User permit (CU), Minor Conditional Use Permit (MCUP), and Expressive Use Permit (EUP) |
| CH-4 | Chatbot – Application Link | The system shall be able to search for common application permits and forms. Such as:   * Short term rental * Home occupation permit * Variance/minor variance, * Conditional Use and Minor Conditional Use * Accessory dwelling unit * Accessory structure * Zoning * Certificate of Appropriateness (CoA) * Tree removal * Reasonable Accommodation Permit |
| CH-5 | Chatbot – Retrieve Regulations | The system shall be able to retrieve regulations. Such as:   * Short term rental * Home Occupation permit * Occupation riding regulation * Occupation license regulations * Accessory structure * Zoning * Home occupation sanitation * Accessory dwelling units * Projects subjected to Certificate of Appropriateness (CoA) * Sober living facility operations |
| CH-6 | Chatbot – Retrieve Standards | The system shall be able to retrieve standards for developing structures. Such as:   * LASP Development Standards * EPSP Development Standards * FGSP Development Standards * Waiver of development standards * Accessory dwelling units * Daycares * Lots |
| CH-7 | Chatbot – Retrieve Permits Applicable to Zoning Statutes | The system shall be able to retrieve permits allowable within a specific zone, based off of user input of the address information. |
| CH-8 | Chatbot – MapQuest integration | The chatbot shall be linked to the MapQuest geocoding API sharing identified geolocation information. |
| CH-9 | Chatbot – Address Requirements | The chatbot requires the following information to retrieve zoning and permit information:   * Street Address * City * State * Zip code |
| CH-10 | Address Validation | The system shall validate the address information against existing zoning and permit information. |
| CH-12 | Chatbot – Other Information | The chatbot shall provide a URL to the website’s frequently asked questions page, if a user submits information not relevant to zones and permits. |
| CH-13 | Chatbot - Help | The chatbot shall display information in using the chatbot when the word “help” is the input. |

2.3 Use Case Diagram

The following diagram illustrates the use cases for the City Chatbot. The primary actor in this case is the city resident who will be searching for request permits and or regulation information as it pertains to a specific address. The customer fulfillment actors in this use case are the Chat dialog service that will facilitate the back and forth response between the user the application, the address verification service that interacts with the system to verify a user’s given address, the mapping service that validates the users address against the city zones, and the city code service that provides the requested information for a given question.

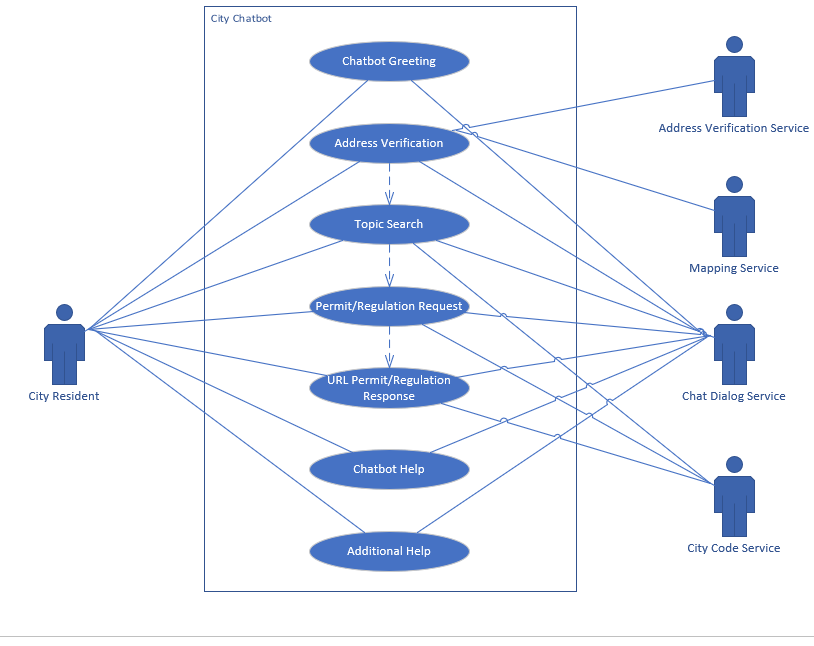


Figure Use Case Diagram of System

2.4 Use Cases

Use cases represent a small workflow specific to the functional requirements of the application. These are the basic use cases of the application.

Table : Allowed Land Use, CH-3

| ID | CH-3, Allowed Land Uses |
| --- | --- |
| Description | The chatbot shall provide zoning and permit links and information. This shall provide the following:   * Definition * Land use category * Variance * Entitlement process where applicable for Conditional User permit (CU), Minor Conditional Use Permit (MCUP), and Expressive Use Permit (EUP) |
| Primary Actor | Regular User |
| Trigger | The user inputs one of the following key words:   * Conditional Use Permit or CUP * Minor Conditional Use Permit, MCUP * Variance * Minor variance * EUP * Expressive Use Permit |
| Pre-conditions | Chatbot is active. |
| Post-conditions | The chatbot shall provide the following information for each trigger:   * CUP or MCUP   <https://library.municode.com/ca/pasadena/codes/code_of_ordinances?nodeId=TIT17_ZONING_CODE_ART6LAUSDEPEPR_CH17.61PEAPDI_17.61.050COUSPEMAPL>   * Variance or Minor variance   <https://ww5.cityofpasadena.net/planning/wp-content/uploads/sites/56/2017/09/Variance-Minor-Variance-Permit-Application.pdf>   * EUP   <https://ww5.cityofpasadena.net/planning/wp-content/uploads/sites/56/2017/09/Expressive-Use-Permit.pdf><https://library.municode.com/ca/pasadena/codes/code_of_ordinances?nodeId=TIT17_ZONING_CODE_ART6LAUSDEPEPR_CH17.61PEAPDI_17.61.060EXUSPE> |
| Exception | The chatbot relays its exception message and link. |

Table : Application Link, CH-4

| ID | CH-4, Application Link |
| --- | --- |
| Description | The system shall be able to search for common application permits and forms for the following:   * Short term rental * Home occupation permit * Variance/minor variance, * Conditional Use and Minor Conditional Use * Accessory dwelling unit * Accessory structure * Zoning * Certificate of Appropriateness (CoA) * Tree removal * Reasonable Accommodation Permit |
| Primary Actor | Regular User |
| Trigger | The user enters key word “application” and one of the following terms:   * Short term rental * Home occupation permit * Variance/minor variance, * Conditional Use and Minor Conditional Use * Accessory dwelling unit * Accessory structure * Zoning * Certificate of Appropriateness (CoA) * Tree removal * Reasonable Accommodation Permit |
| Pre-conditions | The chatbot is active. |
| Post-conditions | The chatbot shall provide the following links for terms:   * Short term rental   <https://www.cityofpasadena.net/wp-content/uploads/sites/30/Supplemental-Forms-for-STR.pdf?v=1599068079137>   * Home occupation permit   <https://library.municode.com/ca/pasadena/codes/code_of_ordinances?nodeId=TIT17_ZONING_CODE_ART5STSPLAUS_CH17.50STSPLAUS_17.50.110HOOC>   * Variance/minor variance   <https://www.cityofpasadena.net/planning/wp-content/uploads/sites/30/Variance-Minor-Variance-Permit-Application.pdf>   * Conditional Use and Minor Conditional Use   <https://ww5.cityofpasadena.net/wp-content/uploads/sites/56/2017/09/Minor-Conditional-Use-Permit-Use-only-permitted-with-conditions.pdf>   * Accessory dwelling unit   <https://ww5.cityofpasadena.net/planning/wp-content/uploads/sites/56/2018/05/ADU-Covenant-and-Agreements-Instructions-091218.pdf>   * Accessory structure   <https://ww5.cityofpasadena.net/planning/wp-content/uploads/sites/56/2017/09/Accessory-Structure-Covenant_Package-NOV-2018.pdf>   * Zoning   <https://ww5.cityofpasadena.net/wp-content/uploads/sites/56/2017/08/Zoning-Permit-Application.pdf>   * Certificate of Appropriateness (CoA)   <https://ww5.cityofpasadena.net/wp-content/uploads/sites/56/2017/09/Certificate-of-Appropriateness-Review-Packet.pdf>   * Tree removal   <https://ww5.cityofpasadena.net/wp-content/uploads/sites/56/2017/09/Tree-Removal-Private.pdf>   * Reasonable Accommodation Permit   <http://ww2.cityofpasadena.net/councilagendas/2007%20agendas/Aug_13_07/9A2.pdf>   * Home Occupation Permit   <https://ww5.cityofpasadena.net/wp-content/uploads/sites/56/2017/07/Home-Occupation-Permit.pdf> |
| Exception | The chatbot relays its exception message and link. |

Table : Retrieve Regulations

| ID | CH-5, Retrieve Regulations |
| --- | --- |
| Description | The system shall be able to retrieve regulations for the following:   * Short term rental * Home Occupation permit * Occupation riding regulation * Occupation license regulations * Accessory structure * Zoning * Home occupation sanitation * Accessory dwelling units * Projects subjected to Certificate of Appropriateness (CoA) * Sober living facility operations |
| Primary Actor | Regular User |
| Trigger | The user inputs “regulation” and one of the following key words:   * Short term rental * Home Occupation permit * Accessory structure * Zoning * Accessory dwelling units * Projects subjected to Certificate of Appropriateness (CoA) * Sober living facility operations |
| Pre-conditions | Chatbot is active. |
| Post-conditions | The chatbot shall provide the following information for each trigger:   * Short term rental   <https://www.cityofpasadena.net/planning/short-term-rental-regulations/>   * Home Occupation permit   <https://ww5.cityofpasadena.net/wp-content/uploads/sites/56/2017/07/Home-Occupation-Permit.pdf>   * Accessory structure   <http://ww2.cityofpasadena.net/councilagendas/2005%20agendas/Jan_10_05/9A2%20ARTICLE%205%20PART%202.pdf>   * Zoning   <https://www.cityofpasadena.net/planning/planning-division/current-planning-and-zoning/>   * Accessory dwelling units   <https://maxablespace.com/adu-regulations/pasadena-adu-regulations/#:~:text=For%20new%20structures%2C%20the%20minimum,requirements%20in%20their%20municipal%20code>.   * Projects subjected to Certificate of Appropriateness (CoA)   <https://www.cityofpasadena.net/planning/planning-division/design-and-historic-preservation/historic-preservation/historic-preservation-documents/>   * Sober living facility operations   <https://www.transitionalhousing.org/ci/ca-pasadena> |
| Exception | The chatbot relays its exception message and link. |

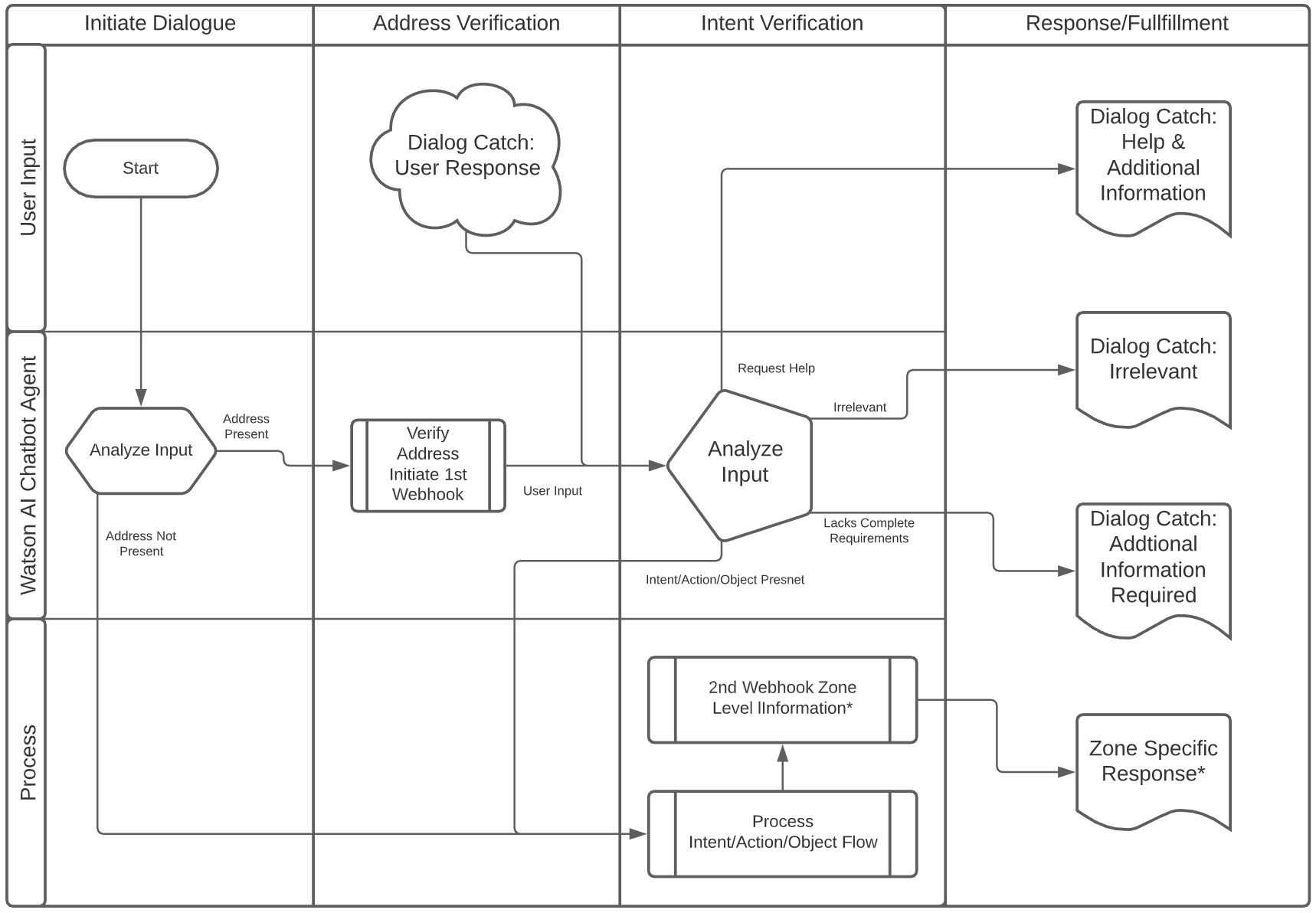
Table : Retrieve Development Standards, CH-6

| ID | CH-6, Retrieve Development Standards |
| --- | --- |
| Description | The system shall be able to retrieve standards for developing the following:   * LASP Development Standards * EPSP Development Standards * FGSP Development Standards * Waiver of development standards * Accessory dwelling units * Daycares * Lots |
| Primary Actor | Regular User |
| Trigger | The user inputs “development standards” and one of the following key words:   * LASP * EPSP * FGSP * Waiver of development standards * Accessory dwelling units * Daycare * Lots |
| Pre-conditions | Chatbot is active. |
| Post-conditions | The chatbot shall provide the following information for each trigger:   * Accessory dwelling units, Daycare, or Lots   <https://library.municode.com/ca/pasadena/codes/code_of_ordinances?nodeId=TIT17_ZONING_CODE_ART2ZODIALLAUSZOECST_CH17.29HIOVDI_17.29.050DEST>   * LASP Development Standards   <https://library.municode.com/ca/pasadena/codes/code_of_ordinances?nodeId=TIT17_ZONING_CODE_ART3SPPLST_CH17.37LIAVSPPL_17.37.060LAGEDEST>   * EPSP Development Standards and EPSP Additional Development Standards   <https://library.municode.com/ca/pasadena/codes/code_of_ordinances?nodeId=TIT17_ZONING_CODE_ART3SPPLST_CH17.32EAPASPPL_17.32.070EPDEST>  <https://library.municode.com/ca/pasadena/codes/code_of_ordinances?nodeId=TIT17_ZONING_CODE_ART3SPPLST_CH17.32EAPASPPL_17.32.080EPADDEST>   * FGSP Development Standards   <https://library.municode.com/ca/pasadena/codes/code_of_ordinances?nodeId=TIT17_ZONING_CODE_ART3SPPLST_CH17.33FAOAANGRSPPL_17.33.050FGDEST>   * Waiver of development standards   <https://library.municode.com/ca/pasadena/codes/code_of_ordinances?nodeId=TIT17_ZONING_CODE_ART4SIPLGEDEST_CH17.43DEBOWAIN_17.43.060WADEST> |
| Exception | The chatbot relays its exception message and link. |

2.5 Chatbot Stimulus Response / Data Flow Diagram

The agent’s stimulus response is best described as the diffusion of customer input through a categorical filter of intents leading to specific fulfillment responses based on customer input. Below is a simplified depiction of the user/agent interaction whose goal is customer fulfillment.

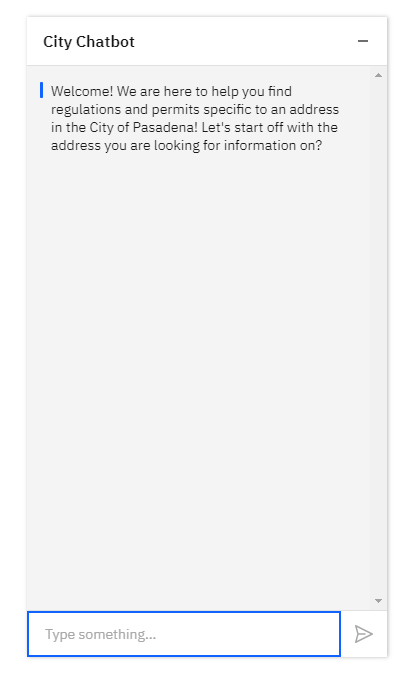
Figure : Data Flow Diagram

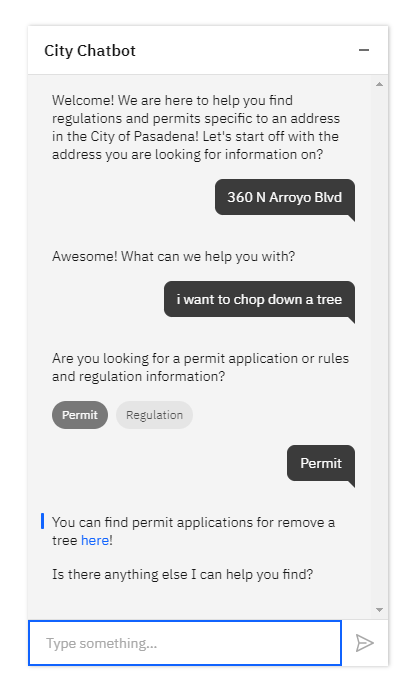


\*If zone is required, the 2nd webhook fulfillment will provide user request relative to matched zone parameters.

**Application Dialog Flow Examples**

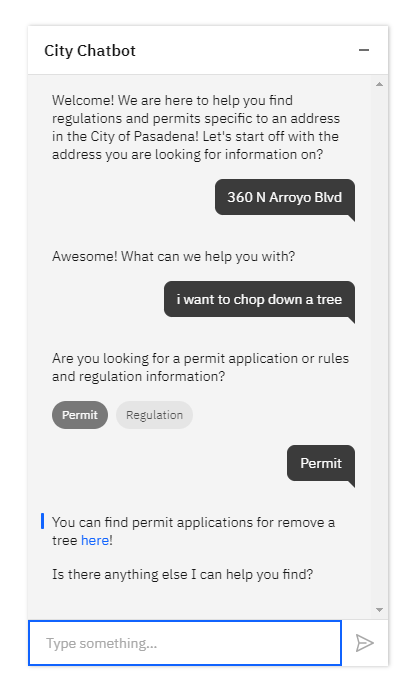
2.6.1 Chatbot – Greeting, CH-1

Description: The agent shall greet the user.

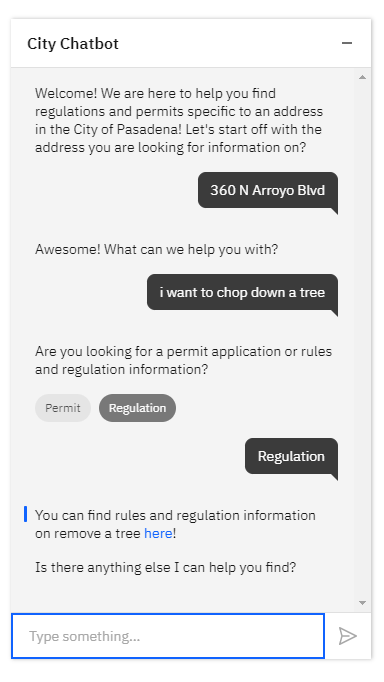
2.6.2 Chatbot – Zoning/Permit Links, CH-2

Description: The agent shall fulfill zone specific permit links.

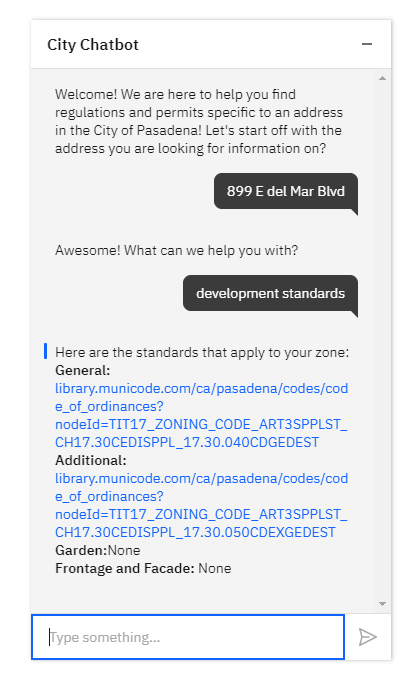
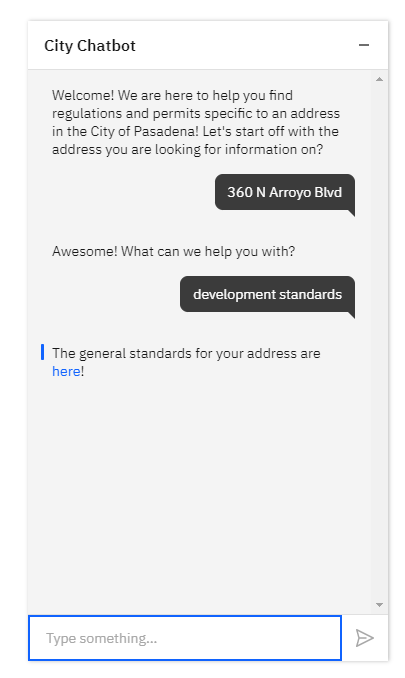
2.6.3 Chatbot – Application Links, CH-3

Description: The agent shall fulfill action and object specific application information in the form of a URL, and when required zone specific relationship.

2.6.4 Chatbot – Retrieve Regulations, CH-4

Description: The agent shall fulfill action and object specific regulatory information in the form of a URL, and when required zone specific relationship.

2.6.5 Chatbot – Retrieve Standards, CH-5

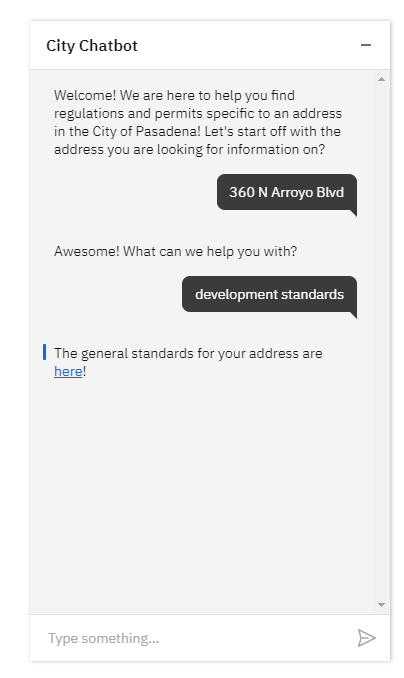
Description: The agent shall fulfill action and object specific standard information in the form of a URL.

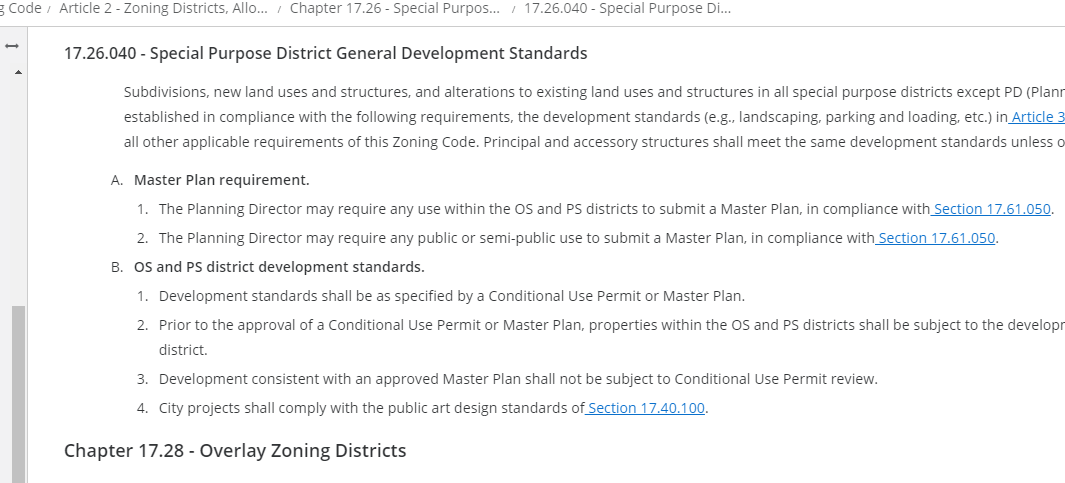
Multiple Standards

Single Standard

2.6.6 Chatbot – Retrieve Permits Application to Zoning Status, CH-6

Description: The agent shall fulfill action, object, and zone-specific permit information in the form of a URL.

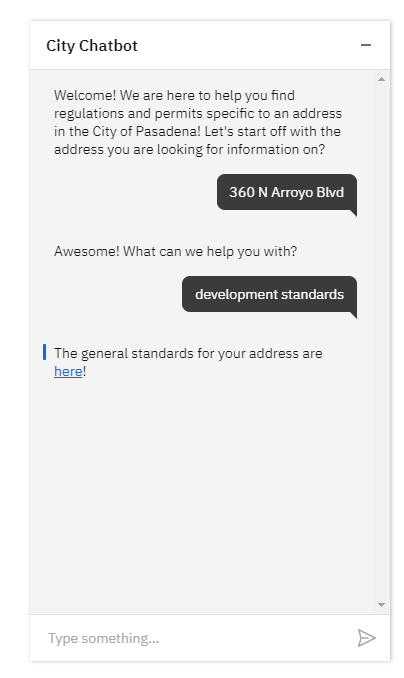




Returned Zone Standard:

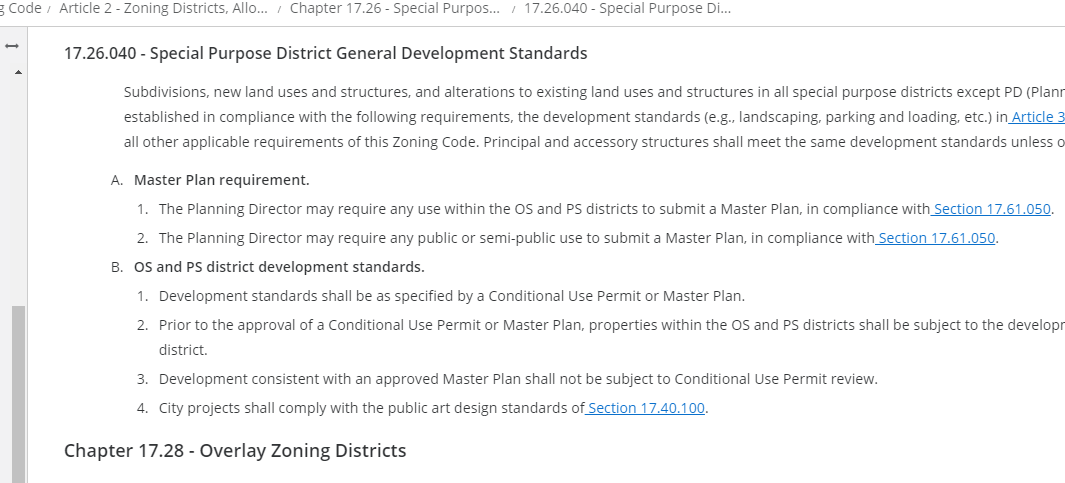
<https://library.municode.com/ca/pasadena/codes/code_of_ordinances?nodeId=TIT17_ZONING_CODE_ART2ZODIALLAUSZOECST_CH17.26SPPUZODI_17.26.040SPPUDIGEDEST>

2.6.7 Chatbot – Chatbot GIS Map Layer Integration, CH-7

Description: The agent shall qualify valid City of Pasadena addresses.

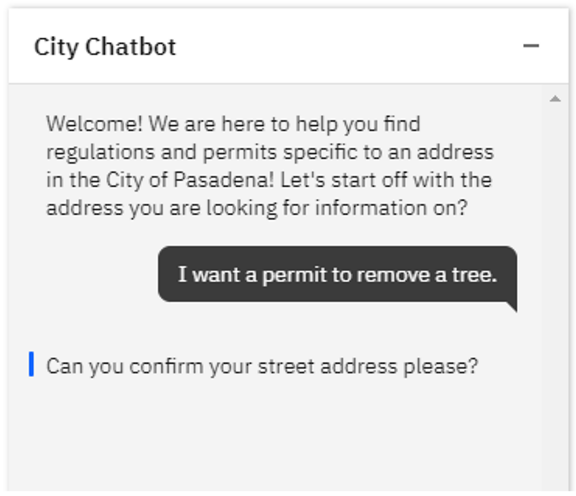
Returned Zone Standard:

<https://library.municode.com/ca/pasadena/codes/code_of_ordinances?nodeId=TIT17_ZONING_CODE_ART2ZODIALLAUSZOECST_CH17.26SPPUZODI_17.26.040SPPUDIGEDEST>



**2.6.8 Chatbot – Address Requirements, CH-8**

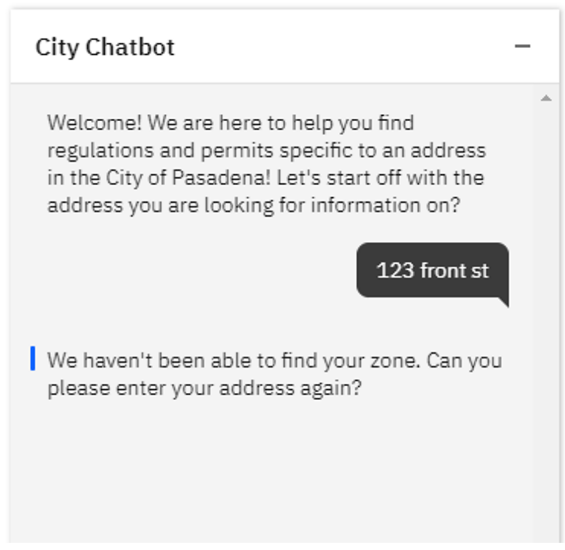
Description: The agent shall qualify input to require an address input when required.



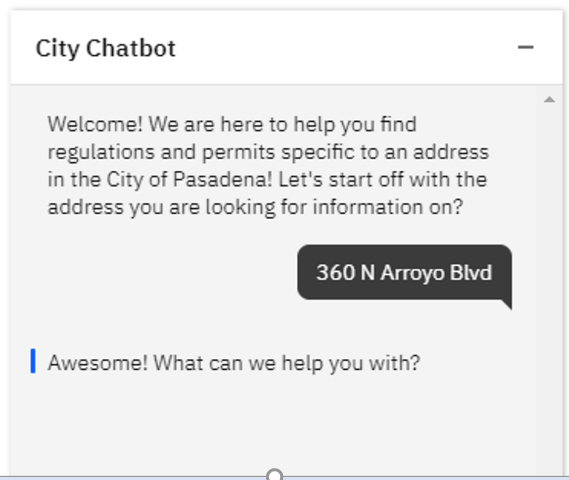
2.6.9 Chatbot – Address Validation, CH-9

Description: The agent shall qualify valid address formats.

Permit Address Required



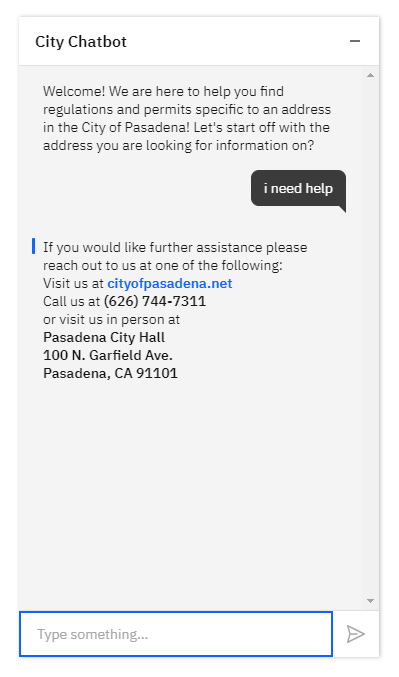
Development Standards Address Required



2.6.10 Chatbot – Other Information, CH-10

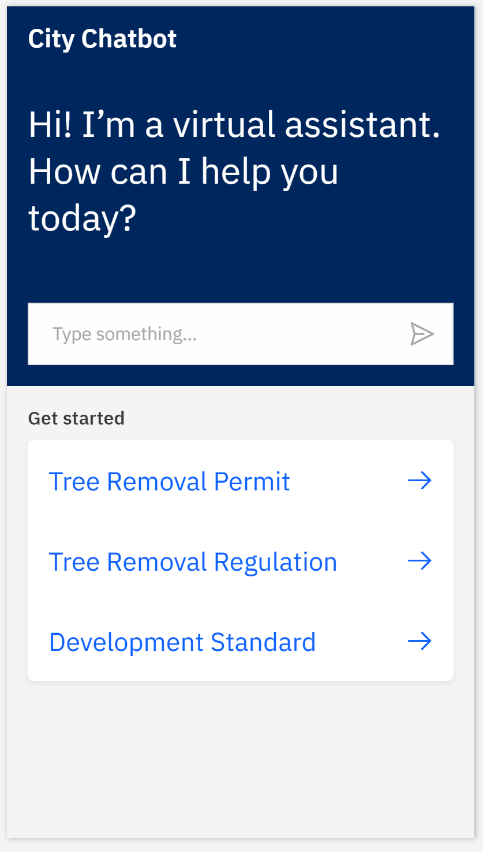
Description: The agent shall prompt when the customer input is irrelevant.

2.6.11 Chatbot – Help, CH-11

Description: The agent shall provide a user Help fulfillment feature.

2.6.12 Chatbot – Get Started, CH-12

Description: The agent shall have a quick link panel displaying the following list options:

* Tree Removal Permit
* Tree Removal Regulation
* Development Standards

Note: See related use case for customer fulfillment requirements.

Figure Diagram of Data Flow in System

2.7 User Classes and Characteristics

User classes and characteristics lists and describes the user base.

Customer/user: A City of Pasadena Zoning and Planning website patron, person, or entity accessing the city of Pasadena’s Zoning and Planning website.

IBM Watson Assistant® - Also referred to as City Chatbot, agent, application or system; is the technical solution providing customer-oriented fulfillment for the City of Pasadena Zoning and Planning body.

2.8 Assumptions and Dependencies

This section describes the assumptions and dependencies of the system. Assumptions inlay foundational principles in which all other logical assumptions can be built upon. Dependencies describe the system as a functional and complete working body of dependent constituent sub-components.

2.8.1 In Scope

* The information called by the chatbot resides in a custom MySQL database used to show functionality.
* The chatbot software shall be open source or free ware, until dictated to change by the primary stakeholder.
* Geocoding shall be established by MapQuest geocoding API.
* The system relies on a stable internet connection.
* Development and testing are conducted on a mixture of local systems and external cloud services.

2.8.2 Out of Scope

* The information called by the chatbot is not directly from the city of Pasadena’s database.
* The system relies on a stable internet connection.
* Development and testing are conducted on a mixture of local systems and external cloud services.
* The system is not configured to execute on the city of Pasadena’s IT infrastructure.

3 Software Requirements

This section outlines the basic software requirements for developmental, testing, and operational objectives. Development includes local and external account and programmatic services conducive to architectural manipulation of the system. Testing tools describe solutions impacting quality assurance of system functionality and goals. Operation objectives reports on basic services and applications required to manage and administrate over the system. Outside of basic desktop, laptop, and operating system requirements. The following is summary of the software requirements for this application:

3.1 Developmental, Software Requirements

* Java Spring.io – A programming and configuration model for java enterprise applications.
* Testing, Software Requirements
* SonarQube – Integrates within workflow providing automated code review, bug detection, vulnerability scans, and code smells.
* Selenium WebDriver – Browser based regression testing automation tool, able to manage suites of testing protocols.

3.2 Operational, Software Requirements

* IbmCloud Services Account – A cloud based virtual services that distributes applications based on advanced data, AI, deep learning, and hybrid cloud services.
* GitHub – A collaboration and version control platform; allowing developers to host and collaborate on projects from anywhere.

3.1 Major System Components

This section outlines the primary components of the application. Which comprise of the IBM Watson Assistant®, custom java Spring.io service layer, MapQuest API, and MySQL database.

IBM Watson Assistant – AI engine that processes natural langue through an intent/entity-based dialog framework. Providing customer support by acting as an intermediary between organizations and customers.

Custom API Service Layer – A Java Spring.io based API that routes incoming and outgoing information between the agent, MapQuest API, and MySQL database, returning request specific information.

MapQuest GeoCoding API – Geo-coding verification service, used to validate City of Pasadena addresses; and to retrieve city specific zone information.

MySQL Database – A commonly used relational SQL database server.

4 External Interface Requirements

This section reports on the customer/user facing portion of the system, often referred to as the graphical user interface (GUI). Clarifying the GUI’s basic features and behaviors within the parameters of user interaction.

4.1 User Interfaces

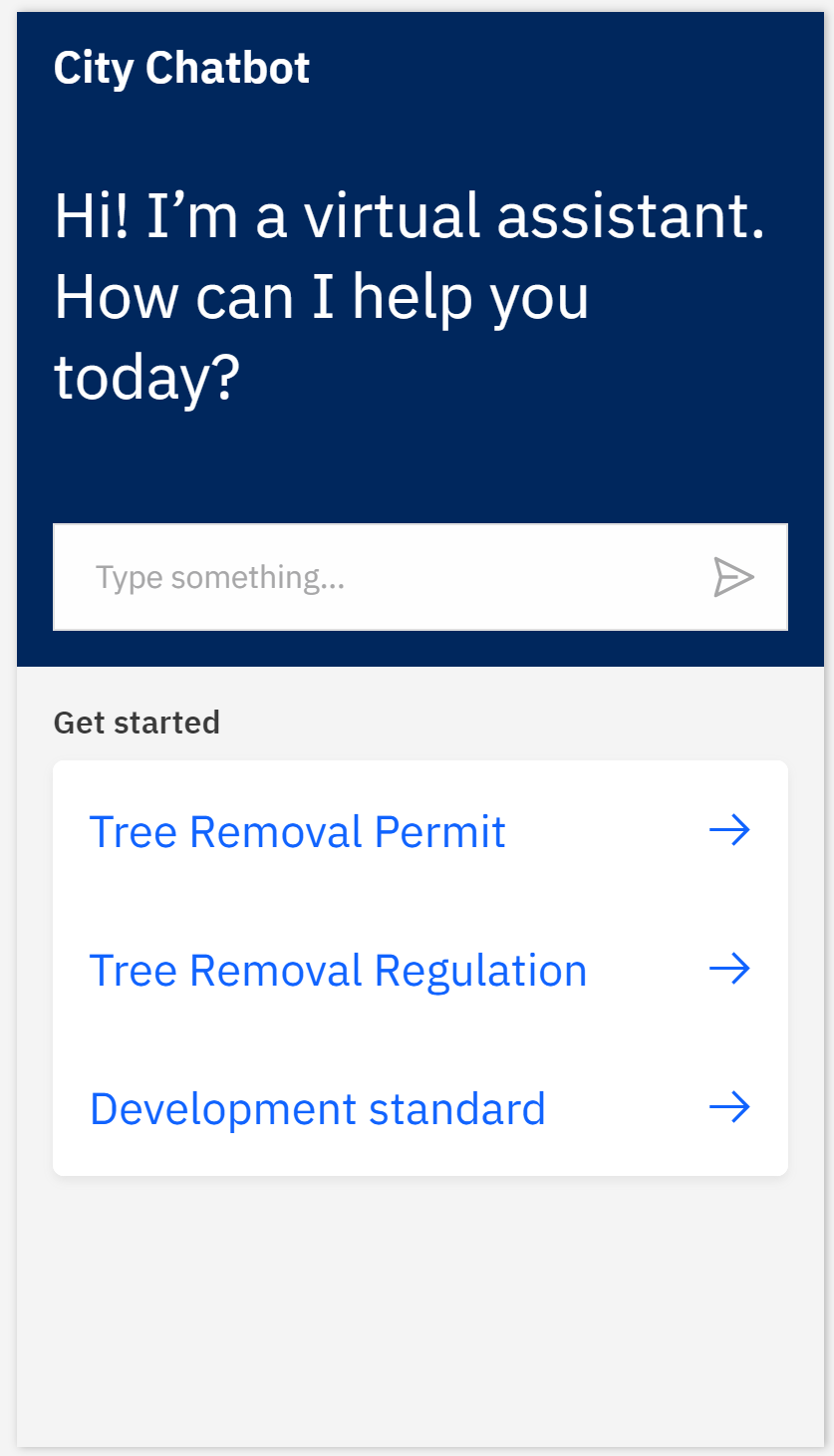
The graphical user interface shall be open source and include address verification using MapQuest GeoCoding API. Below, SnatchBot is used as an example for this document to showcase the basic functionality of chatbots. Chatbots use natural language capabilities to discern the intent of what a user is saying. The chatbot has two graphical objects. One is the inactive chatbot. The second is the active chatbot. The chatbot has an audio input option and uses voice recognition capabilities.

The inactive chatbot is represented by a chat icon that floats in the lower left corner of the browser window. Clicking on the chatbot icon activates the chatbot, and the user is welcomed with a greeting.

Figure Chatbot Start Icon

Residing within a browser, activating the chatbot icon will trigger the chatbot to expand into a sub window within the browser. The chatbot contains the following:

* Text Input Box – The text input box allows users to enter queries into the application for fulfillment.
* Submit Button – The submit button shall commit the text within the text input box. Alternatively, the user can press the “enter” key when completing their entry.
* Get Started Panel. A quick link reference style option, providing a shortcut to a few system features.



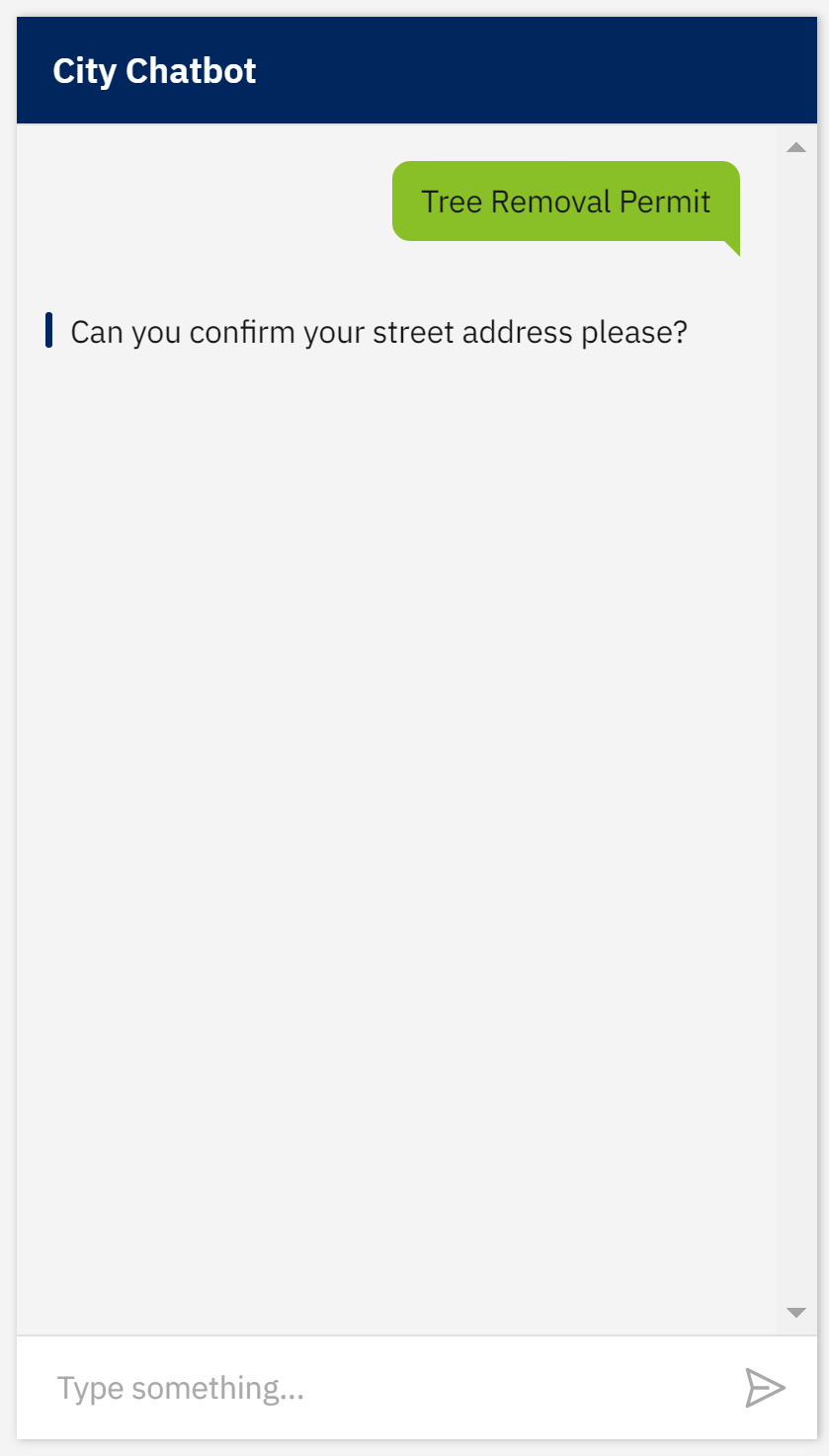
Submit Button

Get Started Panel

Text Input Box

Figure Chatbot Asks User for Input

* Once the chatbot is activated. The user is then able to input their question.



Application Response

Customer Input

Figure User Can Enter Input

4.2 Hardware Interfaces

Usable hardware interfaces are common devices such as desktop computers, laptops, mobile phones, and tablets.

4.3 Software Interfaces

Other than the chatbot application, usable software interfaces are common operating systems residing on end user devices.

4.4 Communications Interfaces

Usable communication interfaces are common telecommunication networks accessing the internet for normal everyday people.

5 Other Nonfunctional Requirements

5.1 Performance

NF-1.1: The chatbot admin application shall not limit the number of accounts that can be created.

NF-1.2: The chatbot shall give a response to the user within 5 seconds.

NF-1.3: The chatbot shall only return permit and legal code information for the zones that it has been configured for.

NF-1.4 The chatbot shall be able to support the same number of users that the city’s website supports.

5.2 Security

NF-2.1: The chatbot admin application shall require a user to log in to their account in order to access any of its features.

NF-2.2: All network communications shall use the HTTPS protocol.

NF-2.3: The chatbot admin desktop application shall salt and hash all passwords before storing them in the database.

5.3 Availability

NF-3.1: The chatbot admin application shall be accessible by users during the hours that the city office is opened except for periods of unplanned downtime.

NF-3.2 The chatbot shall only be accessible on the city’s website.

NF-3.3 The chatbot shall be accessible on the city’s website at any time except for periods of unplanned downtime.

NF-3.4 The chatbot shall be able to communicate with the chatbot platform as long as the platform is not down.

NF-3.5 The chatbot and chatbot admin application shall be available on the web browsers that the city’s website supports.

5.4 Technology

NF-4.1 The chatbot shall use the <insert chatbot platform once decided> for implementing the interactions with the users.

NF-4.2 The chatbot and chatbot admin applications shall use Java for all non-client facing interfaces.

NF-4.3 The chatbot and chatbot admin applications shall use JavaScript, HTML, and CSS for all client facing interfaces.

NF-4.4 The chatbot and chatbot admin applications shall be deployed in Docker containers on the Microsoft Azure platform.

1. Pending Admin Functionality

The chatbot admin application allows the chatbot admins to add, change, and delete the permit, regulation, and zoning information that the chatbot provides to citizens. Admins will be able to link regulations and permits to specific zones. This information is needed for the chatbot to be able to provide zone specific information to citizens about permits and regulations. The chatbot admin application will be a web application. An OpenLayers map will allow admin users to outline the boundaries of a zone and then be able to link applicable regulations and permits to the zone. The permits and regulations will have the necessary links to provide to citizens in order to find more information about them.

1. **Functional Requirements**

Below are the set of functional requirements for the chatbot admin applications. These requirements need to be completed in order for all the actions specified in the stimulus/response sections to be possible.

REQ-1 The city chatbot admin application shall allow admin users to create an account for the application.

REQ-2 The city chatbot admin application shall allow admin users to log in to their account on the application.

REQ-3 The city chatbot admin application shall allow admin users to add, change, and delete permits from the application.

REQ-4 The city chatbot admin application shall allow admin users to add, change, and delete regulations from the application.

REQ-5 The city chatbot admin application shall allow admin users to add, change, and delete zones from the application.

REQ-6 The city chatbot admin application shall allow the chatbot application to retrieve permit, regulation, and zone information in order to provide it to citizens.

REQ-7 The city chatbot admin application shall be accessible from a web browser provided an internet connection is available.

REQ-8 The city chatbot shall have a Get Started Panel, showing the following quick link list options:

* Tree Removal Permit
* Tree Removal Regulation
* Development Standard

1. **Stimulus & Response Sequences**

Below are a series of stimulus and response pairs that the admin user and the chatbot admin application will have. They represent how the admin users will use the different features of the chatbot admin application.

Stimulus: An admin user clicks the option to create an account on the chatbot admin application.

Response: The chatbot admin application presents a form for the admin user to enter in the information for their account and a button to create their account.

Stimulus: An admin user clicks the option to sign into their account on the chatbot admin application.

Response: The chatbot admin application presents a form for the admin user to enter in their account credentials and a button to submit their credentials for authentication.

Stimulus: A logged-in admin user clicks the option to add a new regulation.

Response: The chatbot admin application presents a form for the admin user to enter in the information for the new regulation and a button to save the information.

Stimulus: A logged-in admin user clicks the option to view the regulations in the application.

Response: The chatbot admin application displays all the regulations in the application.

Stimulus: A logged-in user clicks the option to edit a regulation.

Response: The chatbot admin application presents a form for the admin user to edit the information for the regulation and a button to save the information.

Stimulus: A logged-in user clicks the option to delete a regulation.

Response: The chatbot admin application asks the admin user to confirm the deletion and removes the regulation upon confirmation.

Stimulus: A logged-in admin user clicks the option to add a new permit.

Response: The chatbot admin application presents a form for the admin user to enter in the information for the new permit and a button to save the information.

Stimulus: A logged-in admin user clicks the option to view the permits in the application.

Response: The chatbot admin application displays all the permits in the application.

Stimulus: A logged-in user clicks the option to edit a permit.

Response: The chatbot admin application presents a form for the admin user to edit the information for the permit and a button to save the information.

Stimulus: A logged-in user clicks the option to delete a permit.

Response: The chatbot admin application asks the admin user to confirm the deletion and removes the permit upon confirmation.

Stimulus: A logged-in admin user clicks the option to add a new zone.

Response: The chatbot admin application presents a form for the admin user to enter in the information for the new zone, a map to outline the zone, and a button to save the information.

Stimulus: A logged-in admin user clicks the option to view the zones in the application.

Response: The chatbot admin application displays all the zones in the application.

Stimulus: A logged-in user clicks the option to edit a zone.

Response: The chatbot admin application presents a form for the admin user to edit the information of the zone, a map to change the outline of the zone, and a button to save the changes.

Stimulus: A logged-in user clicks the option to delete a zone.

Response: The chatbot admin application asks the admin user to confirm the deletion and removes the zone upon confirmation.