Software Requirements Specifications

March 1, 2017

Zipper

Siddharth Goyal(15UCS140) Shikhar Dalela(15UCS125)
Jeetendra Ajmera(15UCS056) Hemant Verma(15UCS051)

Project work for the Object Oriented Programming (CSE224) course

Table of contents

3.0 Requirement Specifications

3.1 External Interface Requirements

3.2 Functional Requirements

3.2.1 Generic Z-Code Search

3.2.2 Custom Marker Search

3.2.3 Create Generic Z-Code

3.2.4 Create Custom Marker

3.2.5 Add Custom Marker To Existing Z-Code

3.2.6 Modify Existing Custom Marker Address

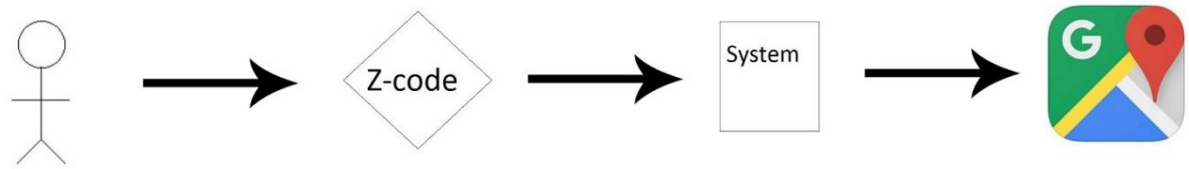
3.2.7 Modify Existing Z-Code Address

3.3 Detailed Non-Functional Requirements

3.3.1 Logical structure of data

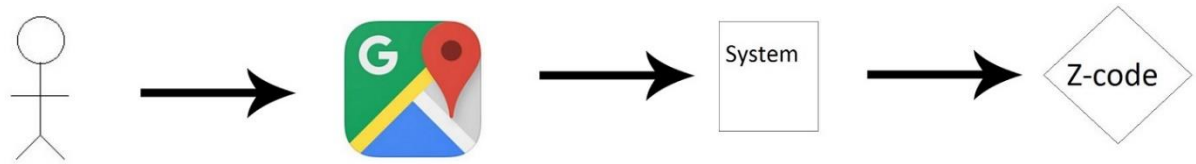
List of figures

Figure 1



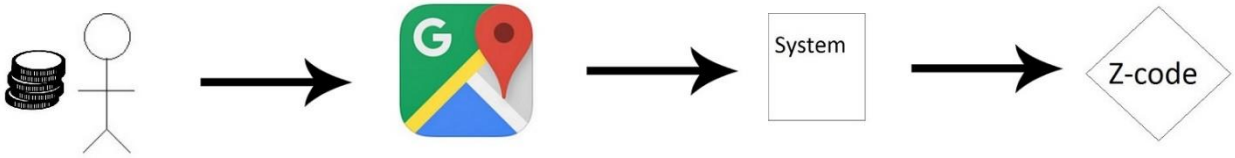
User gives Z-code to the program, it is passed through the system and location marked on Google Maps

Figure 2



User gives location to the program by interacting with Google Maps, the program fetches the coordinates and gives the Z-code generated by algorithm

Figure 3



User gives location to the program by interacting with Google Maps, the user asks for a custom Z-code and pays for it. The program fetches the coordinates and Z-code is generated by algorithm, then the custom Z-code is linked to it.

1.0 Introduction

1.1: Purpose

The aim of the project is to ease out the often-caused troubles of marking a location in real life scenario by making a location pointer k/as ZIPPER. This project would analyses the situation and provide the solutions for the same. It will explain the purpose and used features in the zipper, what will the zipper do in various cases and the constraints under which will the zipper shall work and what shall be its behavior to the external stimuli. The document is to be submitted as a part of the course of Object Oriented Programming.

1.2: Scope of the Project

The software will counter the problem of difficulties faced in hunting a location/place in maps, for instance, any street no or any house number deep down some lane of old city. Many maps don't even show the nearby arenas of the specified location. This would be a very productive way for all the users e any type, to search for his/her given location once the location has been registered in our zipper, that would be provided a unique identification, k/as zipper code with the help of its longitude and latitude, which will act as the tag for that particular location. The software would also be useful for all the companies/or services, which operate on any kind of maps or navigation system such as cab services or delivery hunters by catalysing their process of the location hunting which otherwise manually done could be treacherous and cumbersome. So, for example, the location with "X, Y" longitude might not be shown in the Google maps, but if assigned an alphabetical name to the given location, then it could be easily accessed on any map. To further, increase the utility of the project, we would be providing the custom choices to the given location and then audit it for selling purpose and hence thereby maintaining the competitive environment to survive in the real world economy. Also, the system will also lay scope of generating and maintaining the scope of various custom names, and z-code which has been provided to avoid any collision/overlapping.

1.3: Glossary

Term	Definition
Zipper	Name of the software
Z-Code	The unique identification number generated through a rigorous procedure to be given to the locations
Database	Collection of all the information monitored by this software
User	Any person using the software to navigate
Stakeholder	Any person registering his/her location for a Z-Code.
Software Requirements System	A document that completely describes all the functions of the proposed software and the constraints under which it shall work.

1.4: References

-*https://docs.oracle.com/javase/tutorial/*

- https://www.tutorialspoint.com/java/

-Java the Complete Reference, 7th edition by Herbert Schildt

- (Any more shall be notified later)

1.5: Overview of the Document

The upcoming chapters shall be dealing with detailed analysis, of various scenarios and situations (cases), describing the functionality of the software in various instances.

The next chapter, the Overall Description section describes the informal requirements and establish the context for the technical requirements which is laid in the next chapter.

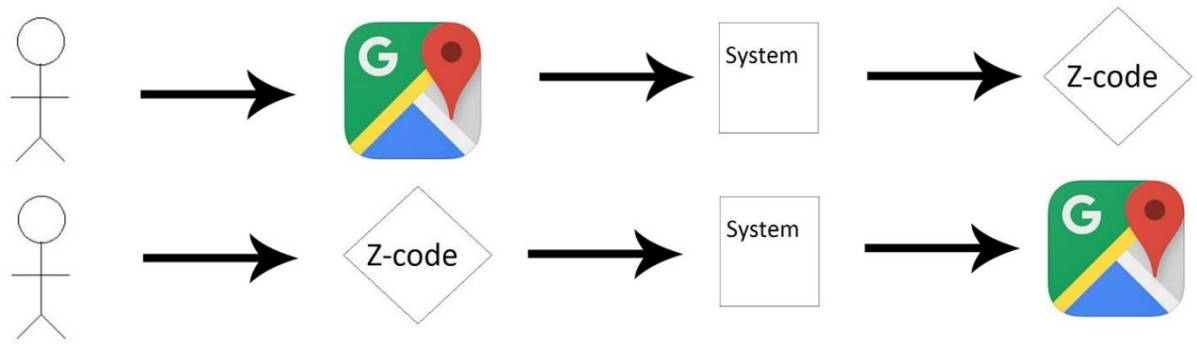
The third chapter, Requirement specification is focussed primarly in the viewpoint for developers and describes the technical terms in details of the functionality and methodology of the software.

Note: Both section of this document describes the same software product in its entirety, but are meant for different audiences thus uses different languages (technical and informal).

2.0 Overall Description

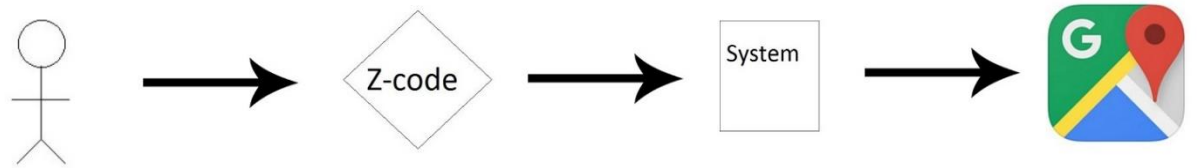
2.1 System Environment

The location pointer system has 2 main user classes, first user, given the Z-code and gets the precise location. The second type of user given the location and takes Z-code. Herein, there is diversification in the type of Z-code a user gets depending on his preferences and system.



2.2: Functional Requirements specification

2.2.1: End user general purpose



Use case: Search Location3

Brief description

The general end case user accesses the program by giving the Z-code through the web interface. Then he/she is directed to the relevant location which will be marked on Google maps.

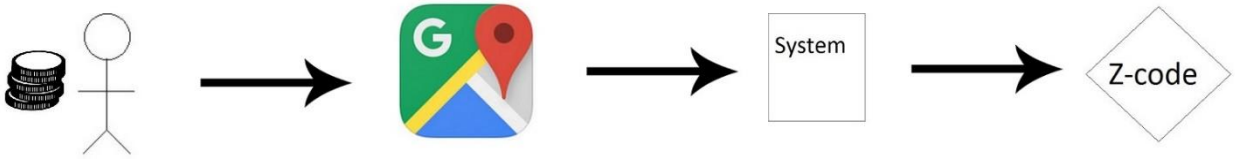
Initial step-by-step description:

When the user uses this case he/she already has the Z-code for the location.

1. Reader enters the Z-code in the interface.
2. The system takes the Z-code and then it is validated against a set of rules.
3. The code is then converted through an algorithm to the co-ordinates.
4. The co-ordinates are passed to Google Maps.
5. The relevant location is then marked on the map.

2.2.2: End Stakeholder Commercial Purpose

A: Government/Public Body



Use case: Make location pointer

Brief description

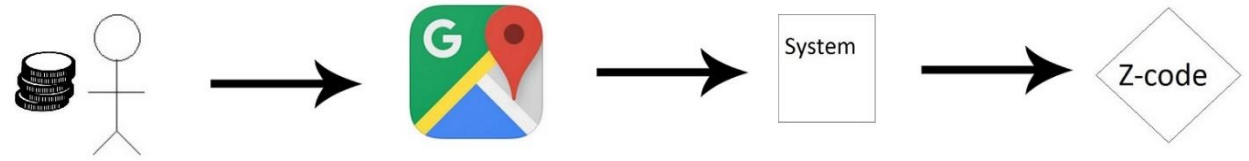
This use case will involve the stakeholder to generate the Z-code for a particular location relevant to her/him. The Z-code will be a generalized one which will be unique to the particular co-ordinate. Further, the stakeholder may use a custom name for the Z-code, provided the availability against a particular amount of monetary/barter exchange. As the case is pertained for a public/government body, the rates shall be subsidized.

Initial step-by-step description: The stakeholder for this case knows the place on the maps.

1. The address yields a rough location of the place.
2. The stakeholder then manually finds the exact place.
3. The system then imports the exact co-ordinates of the location.
4. The co-ordinates are then filtered for the system.
5. The co-ordinates are then converted to Z-code(custom) using the algorithms.

The Z-code (custom) is then displayed to stakeholder which he/she can use for his commercial purpose.

B: Private Body/ Local Business Firm



Use case: Make location pointer

Brief description

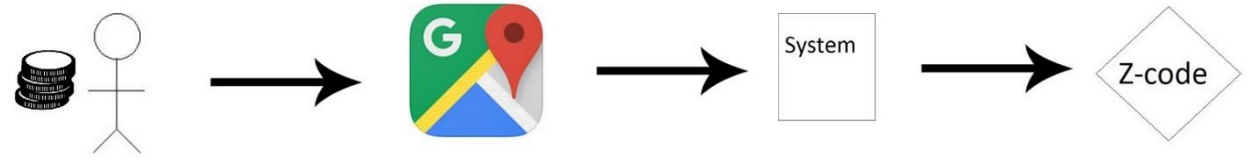
This use case will involve the stakeholder to generate the Z-code for a particular location relevant to her/him. The Z-code will be a generalized one which will be unique to the particular co-ordinate. Further, the stakeholder may use a custom name for the Z-code, provided the availability against a particular amount of monetary/barter exchange or mutual benefits.

Initial step-by-step description: The stakeholder for this case knows the place on the maps.

1. The address yields a rough location of the place.
2. The stakeholder then manually finds the exact place.
3. The system then imports the exact co-ordinates of the location.
4. The co-ordinates are then filtered for the system.
5. The co-ordinates are then converted to Z-code(custom) using the algorithms.
6. The Z-code are relevant to his/her firm which is provided against an amount of money given the availability and competition in the market.

The Z-code (custom) is then displayed to stakeholder which he/she can use for his commercial purpose.

C: Multi National Corporation/Business



Use case: Make location pointer

Brief description

This use case will involve the stakeholder to generate the Z-code for a particular location(s) relevant to her/him. The Z-code will be a generalized one which will be unique to the particular co-ordinate. Further, the stakeholder may use a custom name for the Z-code, provided the availability against a particular amount of monetary/barter exchange or mutual benefits. Further, this case is pertaining to a big business or MNC's which are most likely to have multiple branches in the city, so the custom names shall also be given in light of the same such that a pattern could be generated, for instances, McDonald's for it's multiple branches could be given such as McD:1, McD:2 and so on.

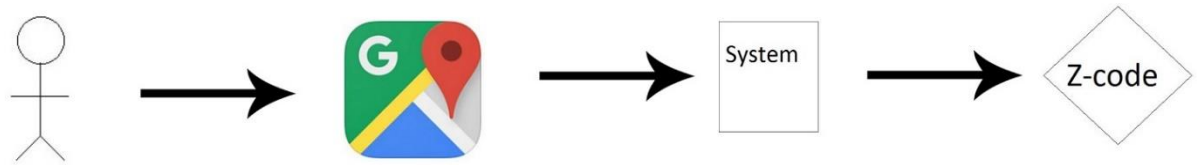
Initial step-by-step description: The stakeholder for this case knows the place on the maps.

1. The address yields a rough location of the place.
2. The stakeholder then manually finds the exact place.
3. The system then imports the exact co-ordinates of the location.
4. The co-ordinates are then filtered for the system.
5. The co-ordinates are then converted to Z-code(custom) using the algorithms.
6. The Z-code are relevant to his/her firm which is provided against an amount of money given the availability and competition in the market.
7. For this particular case, stakeholder most likely shall be requiring Z Code for his/her multiple branches, so the z-code shall be given in a pattern according to his/her requirements.

The Z-code (custom) is then displayed to stakeholder which he/she can use for his commercial purpose.

2.2.3: End Stakeholder non-Commercial Purpose

A: Government/Public Body



Use case: Make location pointer

Brief description

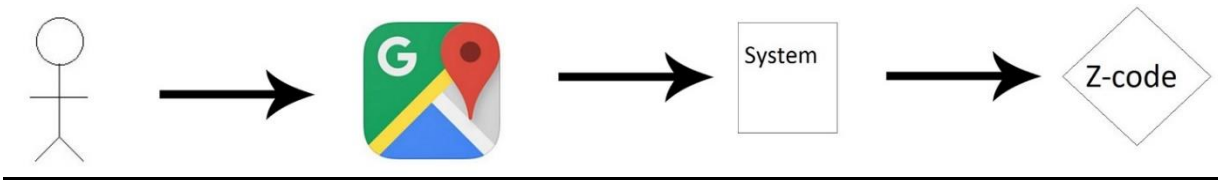
This use case will involve the stakeholder to generate the Z-code for a particular location relevant to her/him. The Z-code will be a generalized one which will be unique to the particular co-ordinate. As the case is pertained for a public/government body, the Z-codes could be provided relevant to the offices names our schemes which is been used through these locations.

Initial step-by-step description: The stakeholder for this case knows the place on the maps.

1. The address yields a rough location of the place.
2. The stakeholder then manually finds the exact place.
3. The system then imports the exact co-ordinates of the location.
4. The co-ordinates are then filtered for the system.
5. The co-ordinates are then converted to Z-code using the algorithms.

The Z-code (custom) is then displayed to stakeholder which he/she can use for his non-commercial purpose.

B: Private Body/ Local Business Firm



Use case: Make location pointer

Brief description

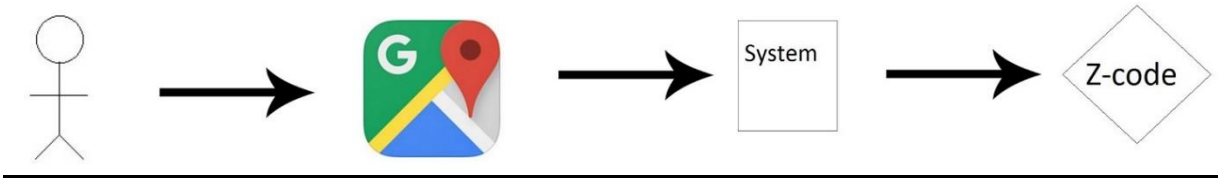
This use case will involve the stakeholder to generate the Z-code for a particular location relevant to her/him. The Z-code will be a generalized one which will be unique to the particular co-ordinate. As the case is pertained for a local business firm(s) the Z-codes could be provided relevant to the market names as far as it do not overlaps with some other stakeholder.

Initial step-by-step description: The stakeholder for this case knows the place on the maps.

1. The address yields a rough location of the place.
2. The stakeholder then manually finds the exact place.
3. The system then imports the exact co-ordinates of the location.
4. The co-ordinates are then filtered for the system.
5. The co-ordinates are then converted to Z-code using the algorithms.

The Z-code is then displayed to stakeholder which he/she can use for his non-commercial purpose.

C: Individual/Residential



Use case: Make location pointer

Brief description

This use case will involve the stakeholder to generate the Z-code for a particular location relevant to her/him. The Z-code will be a generalized one which will be unique to the particular co-ordinate. As the case is pertained for an individual residential, the Z-codes could be provided relevant to the house number as far as it do not overlaps with some other stakeholder.

Initial step-by-step description: The stakeholder for this case knows the place on the maps.

1. The address yields a rough location of the place.
2. The stakeholder then manually finds the exact place.
3. The system then imports the exact co-ordinates of the location.
4. The co-ordinates are then filtered for the system.
5. The co-ordinates are then converted to Z-code using the algorithms.

The Z-code is then displayed to stakeholder which he/she can use for his non-commercial purpose.

2.3: Non-Functional Requirements specification

The system will run on a machine with internet capability. The web interface shall be accessed by the user through a compatible browser.

3.0 Requirements Specifications

3.1 External Interface Requirements

Our project has broadly two external parameters. First and foremost, we will be using the Google Maps API for integration into our system. Google Maps shall be used to graphically mark and depict all the locations. Co-ordinates will be taken/given to Google Maps based on the operation of the program.

Secondly, we will be making a database of all the commercial Z-codes. This will be in order to implement the functionality of custom Z-codes. The table will be used to check/verify for a new/existing Z-code based on user operations.

3.2 Functional Requirements

3.2.1

Use Case Name : Generic Z-Code Search
Trigger : User inputs Z-Code into the search bar
Precondition : The search bar overlay is displayed
Basic Path : 1) The code is checked against a set of rules to verify it as a valid Z-Code
2) If valid, the code is converted to coordinates
3) If the coordinates acquired in the last step have been registered on the
platform already, the Z-Code marker is displayed on Google Maps along with the registered info

3.2.2

Use Case Name : Custom Marker Search

Trigger : User inputs a custom marker into the search bar

Precondition : The search bar overlay is displayed
Basic Path : 1) The custom marker database is searched to find if such a marker has been registered.
2) If the user input marker is matched to an existing one, its accompanying Z-Code is fetched from the database and converted to coordinates
3) Custom Marker is displayed on Google Maps along with the registered info

3.2.3

Use Case Name : Create Generic Z-Code
Trigger : User selects "Z-Code" option from the "New Marker" dropdown
Precondition : "New Marker" button overlay is displayed
Basic Path : 1) The user is redirected to the search bar where he/she is prompted to input their address's street/ closest landmark
2) Google Maps is queried with the input and displays the appropriate result
3) The user is prompted to select their building on the displayed map
4) After building selection is made, a form that asks for details about their address (single/multi-storeyed; commercial/residential) is displayed
5) The building's coordinates are fetched from Google Maps and converted to Z-Code
6) The Z-Code is displayed along with the information provided for the user to confirm
7) Upon user confirmation, the Z-Code along with the info provided is added to the registered Z-Code database
8) A unique access key is generated and displayed to allow authorization in case of future modifications to address details

3.2.4

Use Case Name : Create Custom Marker
Trigger : User selects "Custom Marker" option from the "New Marker" dropdown
Precondition : "New Marker" button overlay is displayed
Basic Path : 1) The user is redirected to the search bar where he/she is prompted to input their address's street/ closest landmark
2) Google Maps is queried with the input and displays the appropriate result
3) The user is prompted to select their building on the displayed map
4) After building selection is made, a form that asks for details about their address (single/multi-storeyed; commercial/residential) is displayed
5) The building's coordinates are fetched from Google Maps and converted to Z-Code
6) The user is prompted to enter a custom marker of their choice
7) The input custom marker is checked against the custom marker database to prevent duplication. If the marker is found to already exist, go to step 6
8) The custom marker, Z-Code and provided info are displayed for the user to confirm
9) Upon user confirmation, the custom marker and info is added to the

custom marker database while the Z-Code and info is added to the registered Z-Code database

10) A unique access key is generated and displayed to allow authorization in case of future modifications to address details

3.2.5

Use Case Name : Add Custom Marker To Existing Z-Code
Trigger : User clicks on "Add Custom Marker" button in address details
Precondition : Address details for a generic Z-Code address are displayed
Basic Path : 1) The user is asked to input unique access key associated with that address
2) Upon authorization, the user is asked to input a custom marker
3) The input custom marker is checked against the custom marker database to prevent duplication. If the marker is found to already exist, go to step 2
4) The custom marker is displayed for the user to confirm
5) Upon user confirmation, the custom marker and info is added to the custom marker database

3.2.6

Use Case Name : Modify Existing Custom Marker Address
Trigger : User clicks on "Modify Address" button in address details
Precondition : Address details for a custom marker address are displayed
Basic Path : 1) The user is asked to input unique access key associated with that address
2) Upon authorization, a form with all the information associated with that address is displayed. All fields are mutable. Form contains confirm button
3) If a change in the custom marker is made, the input custom marker is checked against the custom marker database to prevent duplication. If the marker is found to already exist, an error is displayed and confirm button is greyed out.
4) After making (or not making) changes, the user clicks on confirm
5) Upon user confirmation, the relevant entry in the custom marker database is modified

3.2.7

Use Case Name : Modify Existing Z-Code Address
Trigger : User clicks on "Modify Address" button in address details
Precondition : Address details for a custom marker are displayed
Basic Path : 1) The user is asked to input unique access key associated with that address
2) Upon authorization, a form with all the information associated with that address is displayed. All fields are mutable. Form contains confirm button
3) After making (or not making) changes, the user clicks on confirm
4) Upon user confirmation, the relevant entry in the existing Z-Code database is modified

3.3 Detailed Non-Functional Requirements

3.3.1 Logical structure of data

The structure of data to be stored for commercial Z-codes is as-

Custom Z-code table

Custom Z-code | Algorithm z-code

End