

AFETBILGI SOFTWARE REQUIREMENTS SPECIFICATION

Ceng 350

Cengizhan Deveci - 2448322

Osman Taylan İşleyici

Contents

1	Introduction	5
1.1	Purpose of the System	5
1.2	Scope	5
1.3	System Overview	6
1.3.1	System Perspective	6
1.3.2	System Functions	11
1.3.3	Stakeholder Characteristics	11
1.3.4	Limitations	11
1.4	Definitions	11
2	References	12
3	Specific Requirements	13
3.1	External Interfaces	13
3.2	Functions	13
3.3	Usability Requirements	19
3.4	Performance Requirements	19
3.5	Logical Database Requirements	19
3.6	Design Constraints	19
3.7	System Attributes	19
3.8	Supporting Information	19
4	Suggestions to Improve the Existing System	20
4.1	System Perspective	20
4.2	External Interfaces	20
4.3	Functions	20
4.4	Usability Requirements	20
4.5	Performance Requirements	20
4.6	Logical Database Requirements	20
4.7	Design Constraints	20
4.8	System Attributes	20
4.9	Supporting Information	20

List of Figures

1.1	System Context Diagram	6
1.2	Main Menu in Mobile Devices	7
1.3	Main Menu in Desktop	8
1.4	Maps Integraion	8
1.5	Filter Settings for Map	9
1.6	PDF Download Menu	9
1.7	Accommodation Menu	10
1.8	Healthcare Services Menu	10
3.1	Use Case Diagram	13

List of Tables

3.1	Change language	14
3.2	Contact info and location	15
3.3	Reaching important resources	15
3.4	Location of healthcare services	16
3.5	Reaching other websites	17
3.6	Creating pdf	17
3.7	Filtering information on site by city	18
3.8	Seeing data of website on a map	18
3.9	Contacting with developers	19

1 Introduction

This document is the Software Specification Requirements (SRS) of a website which is afetbilgi.com developed bu a group of METU students and graduates after the Pazarcik Earthquake in February 6, 2023.

1.1 Purpose of the System

Afetbilgi.com is a website which try to deliver accurate information to people. After the Pazarcik Earthquake there was a lot of misinformation on social media platforms and the infrastructure's quality in the earthquake zone was bad. Therefore people who need help were having trouble finding the right information. Thanks to the afetbilgi.com, it delivers the right information with accuracy, speed and simplicity principles.

1.2 Scope

The website is named as afetbilgi.com, the users will be able to reach the important telephone numbers and locations in the disaster situation.

The scope of the system can be listed as

- System is providing users to important locations as a map view, and users can filter the places such as hospitals, food delivery places and temporary accomamodation locations. When it is selected it is navigating by using google maps navigation system.
- System is providing users the all valid active hospitals, evacuation points, safe gathering places and temporary accommodation places in the disaster area to download as a pdf format. Moreover, in the file for all locations, there are how they validate whether the information is correct or not and google maps navigation links.
- System is providing users to select the city where they live so that it filters the information accordingly.
- System is providing users to valid solidarity campaigns, monetary donation links, and blood and stem cell donation places.

1.3 System Overview

This section of the document will provide detailed information about the system with its components.

1.3.1 System Perspective

The purpose of the development of afetbilgi.com is mainly helping people which is affected by disasters like earthquake. For this purpose the website can be used by all people not just limited to people in disaster area. People who are affected can use this application to reach any sort of help. For the other people can find the useful links or locations to help people who are in need. Thanks to this application, a general mobilization can be achieved within the region; therefore, the reach of aid can be accelerated and a wider environment can be easily reached.

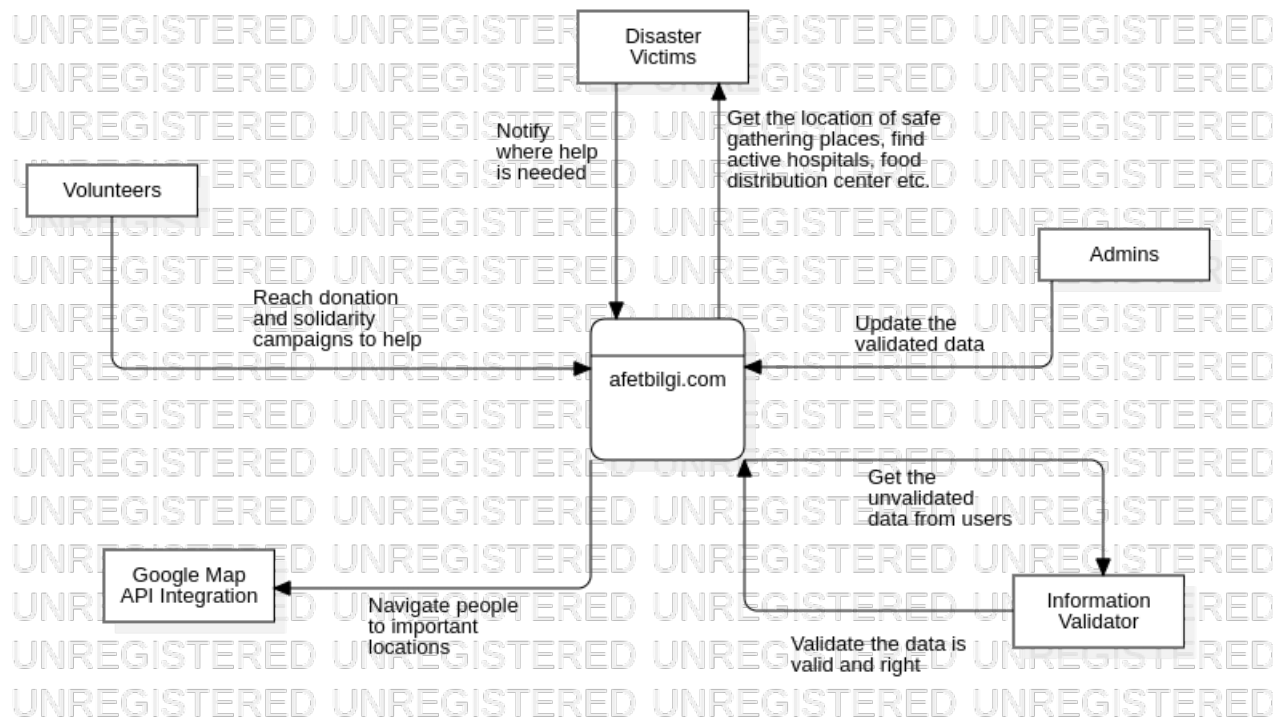


Figure 1.1: System Context Diagram for afetbilgi.com

1.3.1.1 System Interfaces

- **Google Maps API:** Afetbilgi.com uses Google Maps API to navigate people to locations which is on the website to the user. This system shows users to

where they are, close evacuation points, emergency gathering areas, temporary accomodation places, food distribution centers, gas stations, active hospitals and pharmacies. By this integration, in the emergency cases, people can find where and how to go rapidly. Thus, it may increase the survival rate in vital situations.

- **Database Management Interfaces:** Authorized person validate the informations by teams. Afetbilgi.com admins updates the database with the validated information thanks to the validation teams and volunteers. With this mobilization, this system works with accuracy, speed, and simplicity principles. It also prevent disinformation. In addition to that, by the city filtering system, users can reach only the needed information in emergency situations.
- **PDF Integration:** This system allows users that they can download the crucial information for the city they need since the communication and network systems may get damage and it may not reachable. Therefore, by downloading only the crucial information may increase the speed of help.
- **Multi Language Support:** This system allows afetbilgi.com reach the broader effect on the disaster situation. Foreigners in the area can reach the system more easily.

1.3.1.2 User Interfaces

Users can use this application by using their internet browsers. When they reach the website users can see that one of the features of the website is simplicity. All the submenu's and filters are clear and simple. The backend and frontend of the website is lightweight; therefore, in the disaster area users can reach the website with the slow internet speeds.

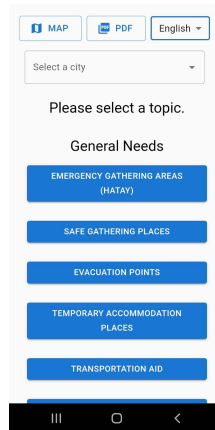


Figure 1.2: Main Menu in Mobile Devices of afetbilgi.com

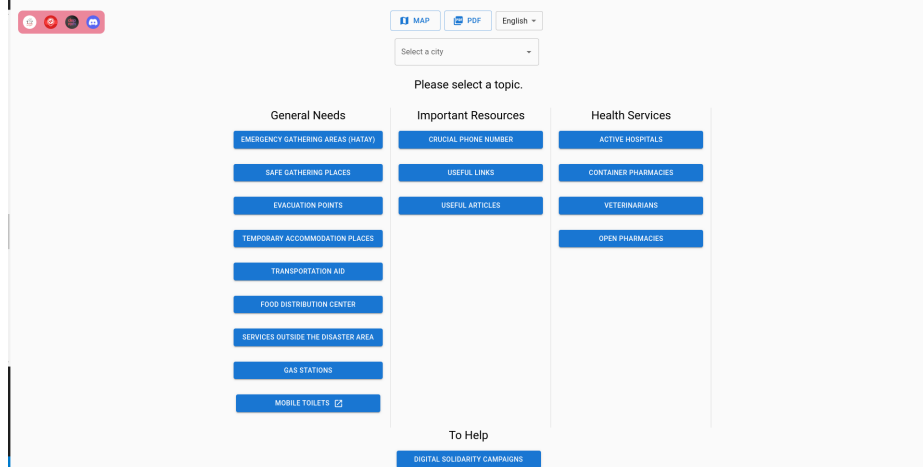


Figure 1.3: Main Menu in Desktop of afetbilgi.com

With the Google Maps API integration, users can see their exact locations. In addition to that, they will find the important locations which they are close to them. Also by the filtering setting, it increases the practicality and becomes task oriented system. When the user select the desired locations, the website redirect them to google maps navigation system.

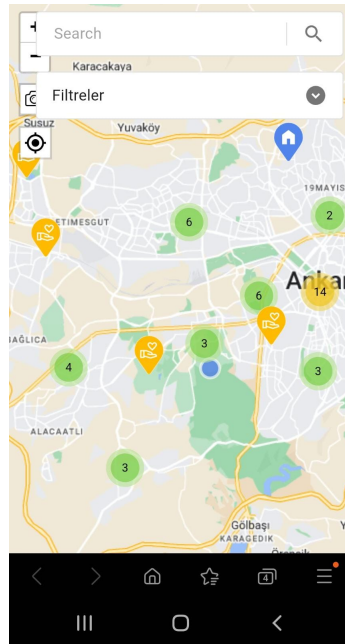


Figure 1.4: Maps Integration

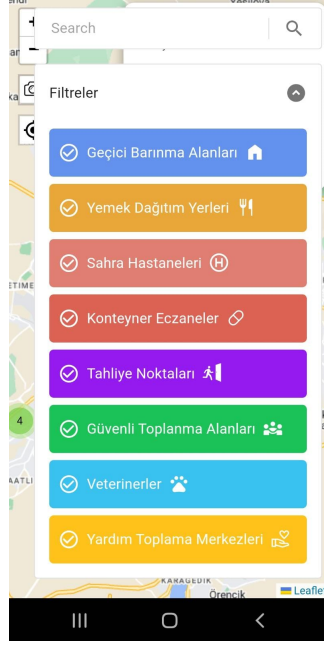


Figure 1.5: Filter Settings for Map

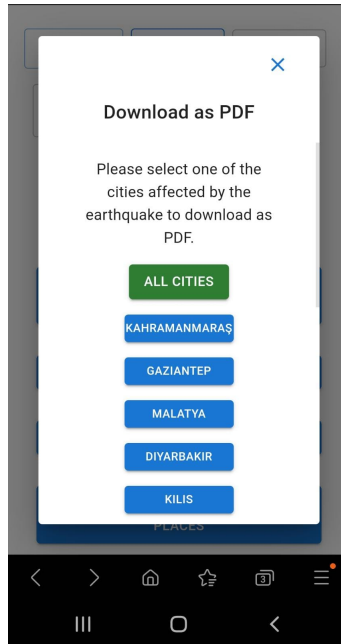


Figure 1.6: PDF Download Menu by Selecting Cities

In which city are you looking for temporary accommodation?

Şehir

Other sources that provide temporary accommodation

Category	Name	Details
	Otelz.com	Link
	Turkish Airlines Holidays	Link
	Ahbab Güvenli Bölgeler Haritası	Link
	HepsiEmlak Dostluk Çatısı	Link
	T.C. MİLLÎ EĞİTİM	

Figure 1.7: Accommodation Menu

Which city are you looking for healthcare services?

ADIYAMAN

HATAY

KAHRAMANMARAS

Active Hospitals

Category	Name	Details
	Deprem Bölgesi Aktif Hastane Listesi	Link
	LÖSEV - LÖSANTE Hastanesi Ücretsiz Hizmet	Link

Last Update:12.04.2023 01:08:03

Figure 1.8: Healthcare Services Menu

1.3.1.3 Hardware Interfaces

The system requires device which has an internet access. If the user want to use Google Maps API it should also have an gps in its device.

1.3.1.4 Software Interfaces

- **Database:** The system uses JSON file to store the data. This system does not require an complicated database system.
- **Operating Systems:** The system can reachable by any device which has an internet browser and access.
- **Google Maps:** The system use Google Maps to show the important locations and where the user are in the map and allow them to reach them.

1.3.1.5 Memory Constraints

There is not an issue about memory constraints in the system. System should have enough memory to hold necessity information however, it requires a very low memory which can sustainable easily.

1.3.1.6 Operations

The operations provided by afetbilgi.com can be partitioned into:

User operations:

-

Admin operations:

-

1.3.2 System Functions

1.3.3 Stakeholder Characteristics

1.3.4 Limitations

1.4 Definitions

2 References

3 Specific Requirements

3.1 External Interfaces

3.2 Functions

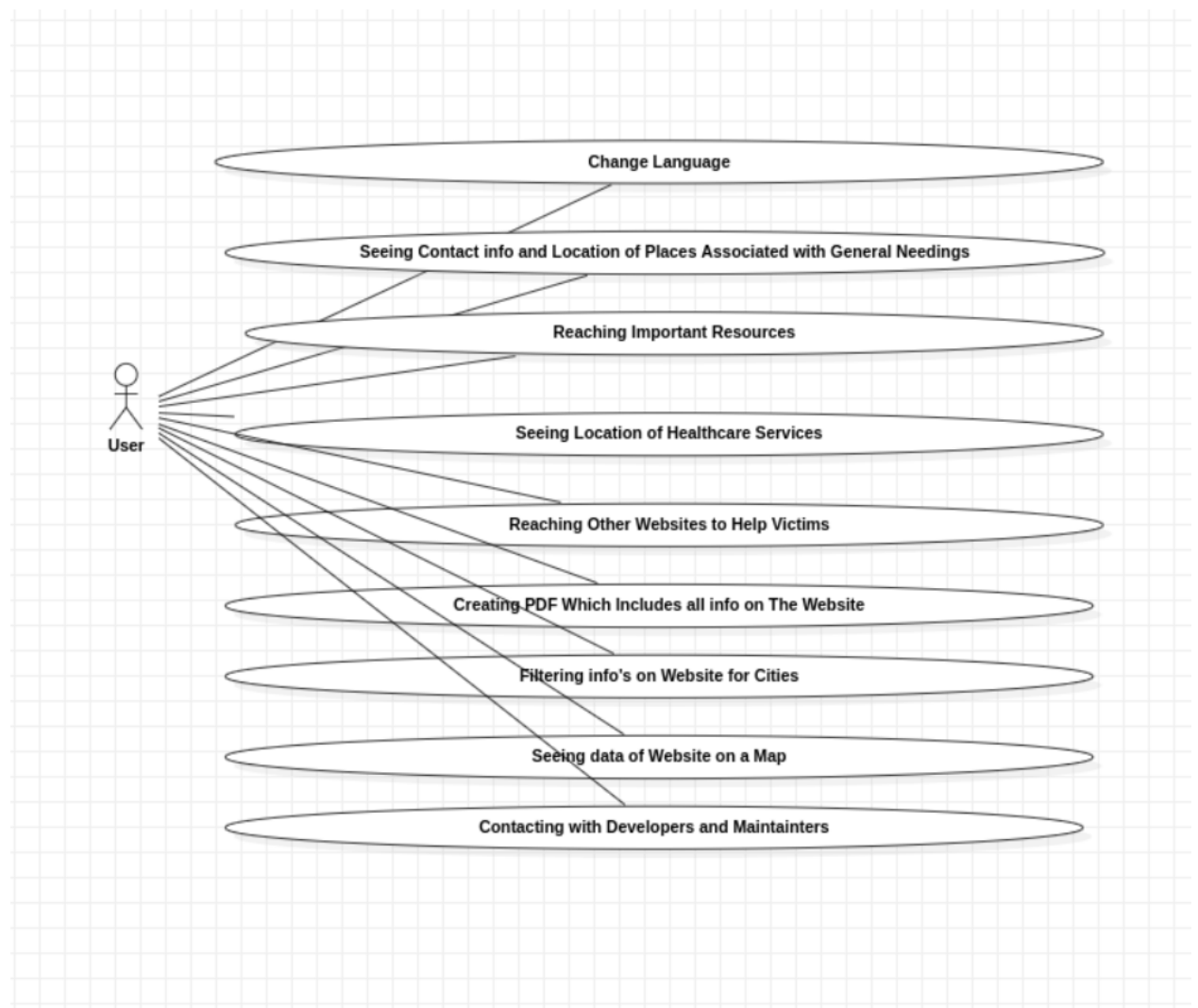


Figure 3.1: Use case diagram.

Use case name	Change language
Actors	User
Description	Users can use change button drop menu to change the language of Afetbilgi.com between Turkish, English, Arabic and Kurdi.
Data	-
Preconditions	User should be in main page.
Stimulus	User tries to change language.
Basic flow	Step 1 - User opens the languages dropdown menu. Step 2 - User select the language. Step 3 - React changes the language of page. With selected one.
Alternative flow	-
Exception flow	If an error is thrown by react.js it's written on browser console.
Postconditions	-

Table 3.1: Change language

Use case name	Seeing Contact info and Location of places
Actors	User
Description	Users can find location and contact info (website, phone number etc.) for their general needs like safe gathering places, gas stations, and evacuation points.
Data	Selected city
Preconditions	-
Stimulus	User tries to get information about their needs.
Basic flow	Step 1 - User clicks one of the eight buttons on main page Step 2 - User selects a city. Step 3 - Site returns the table of suitable locations table. Step 4 - User clicks one of the links to find further info about location.
Alternative flow	Step 1 - User clicks map to open map from main page. Step 2 - User finds the desired location from map. Step 3 - User clicks the location to get further info.
Exception flow	If an error occurs on map side, google map API throws an error.
Postconditions	-

Table 3.2: Seeing contact info and location of places

Use case name	Reaching important resources.
Actors	User
Description	User reach important resources (crucial phone numbers, useful links & useful articles) from the middle panel.
Data	-
Preconditions	-
Stimulus	User tries to get general information about disasters.
Basic flow	Step 1 - User clicks one of the three buttons on main page Step 2 - Site returns table of desired information. Step 3 - User can call the desired phone or go to desired website.
Alternative flow	-
Exception flow	-
Postconditions	-

Table 3.3: Reaching important resources

Use case name	Seeing location of healthcare services.
Actors	User
Description	User can find get the location and some other information about health services (hospitals, pharmacies, veterinarians) from website.
Data	Selected city, (if there is).
Preconditions	-
Stimulus	User tries to get information about health services based on their needings.
Basic flow	<p>Step 1 - User clicks one of four buttons from the right frame.</p> <p>Step 2 - Site returns a general table if there is no city selected from main menu. If there is a selected city site returns location links of services in this city.</p> <p>Step 3 - If user didn't filter cities from main menu, now he/she can.</p> <p>Step 4 - User can reach the location by clicking the location button.</p>
Alternative flow	- Step 1 - User can get the location link from map, by filtering by category, by searching or by finding it in map by hand.
Exception flow	-
Postconditions	-

Table 3.4: Location of healthcare services

Use case name	Reaching other websites to help victims.
Actors	User
Description	User can reach other websites for donating money, blood, stem cell.
Data	-
Preconditions	-
Stimulus	User tries to find places or websites to help victims of earthquake.
Basic flow	User clicks one of the 5 buttons on bottom frame of website.
Alternative flow	-
Exception flow	-
Postconditions	-

Table 3.5: Reaching other websites

Use case name	Creating pdf which includes all info on the website.
Actors	User
Description	User can create a pdf containing the information on website to reach the info offline, or any other purpose.
Data	Selected city. (If there is one).
Preconditions	-
Stimulus	User tries to get all information of website.
Basic flow	Step 1 - User clicks download pdf button at the top of the page. Step 2 - Site returns a preformed PDF containing all information about the city (if selected) selected by the user, in the selected language.
Alternative flow	-
Exception flow	-
Postconditions	-

Table 3.6: Creating pdf

Use case name	Filtering info's on website for cities.
Actors	User
Description	User can filter menus and info's on website to include a selected city.
Data	Selected city.
Preconditions	User should select a city from main menu.
Stimulus	User tries to find relevant information about a city.
Basic flow	Step 1 - User selects a city from main page. Step 2 - Site filters the main menu. (For example if there is no veterinarian in selected city, user removes the veterinarians button.) Step 3 - If user goes another page in website, site continues to filter information for the selected city.
Alternative flow	-
Exception flow	-
Postconditions	-

Table 3.7: Filtering information on site by city

Use case name	Seeing data of website on a map.
Actors	User
Description	Users can see all locations on website visually in a map and use the map to easily find needed services based on their needs and location.
Data	-
Preconditions	-
Stimulus	User tries to find locations easily.
Basic flow	Step 1 - User clicks map button at the top of the page. Step 2 - Site redirects to a map built using Google Map API. Step 3 - User clicks locations or balloons to navigate through map.
Alternative flow	-
Exception flow	-
Postconditions	-

Table 3.8: Seeing data of website on a map

Use case name	Contacting with developers and maintainers.
Actors	User
Description	Users can find contact info and links to source code and social media accounts of developers from a page.
Data	-
Preconditions	-
Stimulus	User wants to reach developers.
Basic flow	<p>Step 1 - User clicks About Us / Contact button at the bottom of the page.</p> <p>Step 2 - Site redirect to a about us page with an Instagram, a twitter and a GitHub link.</p> <p>Step 3 - User can click on of these three buttons or click the mail address to send a mail to developers.</p>
Alternative flow	-
Exception flow	-
Postconditions	-

Table 3.9: Contacting with developers

3.3 Usability Requirements

3.4 Performance Requirements

3.5 Logical Database Requirements

3.6 Design Constraints

3.7 System Attributes

3.8 Supporting Information

4 Suggestions to Improve the Existing System

4.1 System Perspective

4.2 External Interfaces

4.3 Functions

4.4 Usability Requirements

4.5 Performance Requirements

4.6 Logical Database Requirements

4.7 Design Constraints

4.8 System Attributes

4.9 Supporting Information