University of Nairobi

School of Engineering

Department of Geospatial Engineering

A WEBGIS BASED VEHICLE PARKING MANAGEMENT SYSTEM: CASE STUDY OF UPPER HILL, NAIROBI

BY

MWONGELA DOMINIC MATHINA

F56/35272/2019

SUPERVISOR: PROF G.C. MULAKU

A project report submitted in partial fulfillment of the requirements for the Degree of Master of Science in Geographic Information Systems, in the Department of Geospatial and Space Technology of the University of Nairobi

September 2021

This research is submitted with my approval as university supervisor

Supervisor: Prof. G.C. Mulaku

Signature……  ……. Date…16th September 2021…: ……

Declaration of originality

Name of student: MWONGELA DOMINIC MATHINA
Registration: F56/35272/2019
College: CAE
Faculty/School/Institute: ENGINEERING
Department: Geospatial and Space Technology
Course Name: Project
Title of work: DEVELOPING A WEBGIS BASED VEHICLE PARKING MANAGEMENT SYSTEM

1) I understand what plagiarism is and I'm aware of the university policy in this regard
2) I declare that this project is my original work and has not been submitted elsewhere for examination, award of a degree or publication. Where other works or my own work has been used, this has properly been acknowledged and referenced in accordance with the University of Nairobi's requirements
3) I have not sought or used the services of any professional agencies to produce this work
4) I have not allowed, and shall not allow anyone to copy my work with the intention of passing it off as his/her work
5) I understand that any false claim in respect of this work shall result in disciplinary action in accordance with University of Nairobi anti-plagiarism policy

Signature: 

Date: 15th September 2021

Dedication

I dedicate this work to God first for giving me health to read and work tirelessly. Second to my parents, brothers and sisters for their moral and material support accorded not only during this project but also the entire academic journey. Third is to the University of Nairobi who offered scholarship for the entire period and also the Department of Geospatial and Space Technology for guidance and support.

Acknowledgements

First am most grateful to God for the gift of life and health to pursue the Master of Science in Geographic Information Systems. I would also like to acknowledge the University of Nairobi which provided financial support for the entire duration of the programme. Special gratitude to the Department of Geospatial and Space Technology staff who having on my side since bachelor's degree.

I greatly appreciate Ramani Geosystems for willingness to provide datasets for the proposed study. I would like to thank my colleague Titus Chepkwony for suggestions offered during the study. Special thanks to Faith Kendi for helping extract the datasets from different sources. In a special way I would like to appreciate Prof. Mulaku's invaluable contribution and critique as well as provision of valuable information on how the technical design of the system should look like.

Abstract

In an effort to address challenges that face urban transportation, technological innovations that put cities towards sustainable smart cities are imperative. Nairobi County faces enormous challenges in offering efficient parking services to its residents. Some of these challenges include lack of information on available parking lots and also the most convenient routes to navigate to those lots.

To solve these problems, technology reviews of implemented systems across the globe needed be carried out. Potential application of GIS in car parking solutions was studied in detail. Present systems were found from documented literature to be majorly focused on Internet of Things (IoT) with few systems having included capabilities of GIS in their implementation.

The gaps inherent in the present systems were identified and incorporated in the design of a vehicle management system that not only looks at the bird's eye view of the parking lots, but also focuses on the spatial location of the individual parking lots to incorporate user preferences.

This study concludes that, utilizing GIS analytical and visualization capabilities will help drivers locate individual preferred parking lots easily and navigate to those lots.

The study recommends implementation of a webGIS based vehicle parking management system that would be a market place for willing parking lots' owners to have their parking lots put online for any user to access them and thereby generate revenue.

Table of Contents

List of Figures

LIST OF TABLES

ACRONYMS

CBD:	Central Business District
CSS:	Cascading Style Sheets
FHA:	Federal Highway Administration
GDAL:	Geospatial Data Abstraction Library
GEOS:	Geometry Engine Open Source
JKUAT:	Jomo Kenyatta University of Agriculture and Technology
JICA:	Japan International Cooperation Agency
JSON:	JavaScript Object Notation
IBM:	International Business Machines Corporation
MVC:	Model, View and Controller
OGR:	OpenGIS Simple Features Reference Implementation
PGIS:	Parking Guidance Information Systems
SQL:	Structured Query Language

1. INTRODUCTION

1.1. Background

In a study carried out by the International Parking Institute in 2018, an average of 45% of respondents said that real-time information on pricing and availability of parking lots was among the emerging trends in parking (IPI, 2018). Technologies that enhance access control of parking lots, payment and demand for intelligent parking guidance systems that help drivers locate parking spaces efficiently were identified as emerging trends in the industry. Information on availability of parking spaces, public transport information will eradicate time spent roaming around blocks in search of one parking slot.

A parking is a designated area intended for parking vehicles. The vehicles are normally parked for a time ranging from minutes to hours. Finding parking spaces currently involves driving around looking for unoccupied spaces which may turn out to be far from the convenient destination of the driver.

The advent and advancement of technologies such as the internet and World Wide Web (WWW) have provided the platforms from which technologies can be built on to provide solutions to problems such as parking inefficiencies. Geographic Information System (GIS) based on the web utilizes these technologies to create, process and disseminate information both spatial and non-spatial.

In proposing systems that address these trends, a review of information covered under various urban transportation studies is imperative. Review of technologies that have been deployed to address these trends such as Smart Parking Management Systems and Intelligent Transportation Systems (ITS) are reviewed in detail. A Geographic Information System (GIS) based system to manage parking has been suggested as a system that can work as standalone or a tool in integrating the aforementioned two technologies.

GIS technologies bring on-board the unique power of visualization and geographic analysis offered by maps which is intuitive to the users. This is enabled by integration of database operations such as query and statistical analysis of spatial data which is considered special in comparison with other datasets. These capabilities brought by Geographic Information System distinguish it from other information systems by making it widely available to public and private enterprises to model and predict phenomena and plan strategies. The data into the GIS will be generated by the users of the system, a phenomenon known as user-generated content (UGC). UGC is a situation in which content displayed on a system or service is created by the users of that system or service. The system in this case will employ a special type of UGC called Volunteered Geographic Information (VGI) which is the harnessing of various spatial tools to create, organize systematically and disseminate geographic data provided voluntarily by individuals.

1.2. Problem Statement

Parking data like many other datasets have a spatial component. Therefore, use of Geographic Information System (GIS) would be of great importance in providing necessary information to support decision-making by both authorities and drivers. One of the biggest challenges facing urban transport is finding and accessing a parking lot with least time and hassle. Nairobi city being a business hub faces parking issues which have been identified as follows:

- Poor use of available parking spaces
- Business parking spaces taken by commuter parking
- Time and gas used to find open parking space
- Location information of unoccupied parking lots
- 'Parking boys' who illegally control utilization of parking lots

With modern developments in geospatial technology in particular management of data, high processing power and visualization, it is now possible to query, display, analyze and locate parking lots not occupied in real time. The unoccupied lots can be visualized on a map including the best routes to access the lots.

The proposed system intends to solve the above parking issues by relaying real-time information to drivers on the status of parking lots utilization and also the location of unoccupied lots. Drivers can book parking lots in real-time, the information generated there-in will be used to show occupied and unoccupied parking spaces.

1.3. Objectives of the study

1.3.1. General Objective

To develop a web based open air parking management system for the study area.

1.3.2. Specific Objectives

- To digitize parking lots from an aerial photograph
- To design a spatial database for parking lots.
- To develop a web application for accessing the database.

1.4. Justification of the Study

The parking management system currently used in Nairobi County has undergone some digitization efforts especially in terms of e-payment. These efforts however have been mainly to boost revenue collection and reduce the levels of corruption in the sector. The driver's issues in trying to comfortably locate a parking lot at their convenience have not been addressed to date. This means that drivers still have to drive around and locate convenient parking lots.

Nairobi CBD being a business hub is an ideal place to develop and test this kind of system. Due to traffic congestion and the number of vehicles that require parking lots on a daily basis, success of the system in Nairobi CBD means that the system can be replicated in any other place.

The system intends to help drivers find parking spaces in real time and navigate to those spaces without having to physically look for spaces which may not even be convenient to them in the first place. The proposed system aligns with the 11th goal of the sustainable development goals which envisions sustainable cities and communities.

1.5. Scope of the Study

The scope of the study will be limited to development of a webGIS open air Parking Management System in Nairobi Upper Hill area. The system will aid in visualization of the parking lots and state of occupancy. The information generated from the system will support decision making in locating parking spaces. The study will only look at a small part of the CBD which is the Upper Hill area. The parking type involved in this study would be open-parking spaces and off-street parking.

1.6. Organization of the report

This research study is organized into five chapters. Chapter One gives an introduction to the study by providing the background information, outlines the objectives of the study, provides the study justification and scope of the study. Chapter Two provides detailed overview of the underlying literature such as existing car parking technologies, parking situation in Nairobi, technologies that support the proposed system, the gaps that exist in the current vehicle parking management system and case studies where similar studies have taken place. Chapter Three gives detailed overview of the methodology adopted to design the vehicle parking management system. Chapter Four presents the results and the analyses which includes various functionalities of the system. Chapter Five gives conclusions and recommendations looking into the future opportunities. This is followed by a references section.

2. LITERATURE REVIEW

2.1. Parking Situation in Nairobi

A study carried out by Katahira Engineers International, (2005) in the Nairobi CBD found out that double or triple parking of on-street parking was because of maximum occupancy of parking lots being higher than the available number of slots. The study found out that the demand for off-street and parking in the buildings was found to be lower than the supply. The average parking time in terms of the parking type carried out is as shown in Table 1.

Table 1: Average Parking Time

	On-street parking	Off-street parking	Building parking	Total average
Average Parking Time (minutes.)	100	246	245	178

The study found that the trip purpose of parking users was largely work, business and private polling at average 46%, 29% and 18% respectively with the remaining 7% shared between school and other purposes. The survey found out that most users prefer an average walking time of 5 minutes from the parking lots to their destination. The parking spaces available in 2005 were as follows, the demand for parking lot according to the survey found an increase rate of 1.9% in a span of 6 years, which corresponds to the increase rate of the traffic generated over the same period.

Katahira Engineers International (2005) found that theoretically, total car parking supply in the CBD is commensurate with demand despite the imbalance that exists by the type of car park as shown in Table 2.

Table 2: Parking space and utilization

	On-street	Off-street	Building	Total
Car Parking Space	3,941	3,834	7,089	14,864
Utilization (%)	140	95	50	

The study pointed out that maximum usage of the car parks was still an issue primarily due to the time it takes driving around to look for a vacant parking slot. Introduction of a Parking Guide System was among measures recommended to maximize utilization of available parking space and decrease the amount of traffic caused by drivers driving around at slow speeds looking for vacant parking spaces in the busy CBD. A study by JICA Study Team (JST) also revealed that cars that cannot park, together with waiting taxis and *matatus* hinder free traffic movement in major roads and also streets in the CBD (JICA, 2006). For this reason, construction of multilevel parking is inevitable. Most of the traffic congestion in Nairobi CBD results from on-street parking which may be reduced by having on-street lots located on major streets (Abdi , 2019).

The Nairobi County Integrated Development Plan (2018-2022) identifies parking fees as one of the major revenue streams at 17% and second among the major five revenue streams (Nairobi City County, 2014). For example, in the financial year, 2016/2017 parking fees generated 2.25billion shillings against a target of 5.5billion shillings. To reduce this deficit and reduce traffic congestion in the city, the development plan identifies the following strategic objectives among others:

i. Introduction of e-payment system
ii. Installation of traffic/parking information systems
iii. Automation of services using tools such as GIS for the greater convenience of citizens.

2.2. Internet and Web GIS

The adoption and integration of the internet in nearly every day's activity worldwide, online surfing, e-commerce, virtual interactions, taxicab services and Internet of Things (IoT) has made the Internet a basic necessity of the modern society.

The exponential internet infrastructure development has changed how information is accessed, exchanged and processed (ESRI, 2013). Likewise, GIS data, which is often complex and voluminous, has become easy to access, share and manipulate, and will get better with deployment of 5th generation wireless technology.

Internet GIS is an application area that utilizes the internet and other internet working systems to facilitate access, processing, organizing and distribution of geographic information and spatial analysis knowledge (Zhong-Ren & Ming-Hsiang , 2003). GIS processing on the internet is today a reality as demonstrated by platforms such as Google Earth Engine.

Web GIS is a type of distributed information system, comprising a server that enables dynamic discovery of information and a client that facilitates access of information, where the server is a GIS server and client is a web browser, desktop application or mobile application. Web GIS can be viewed as a subset of internet GIS using the World Wide Web as a primary means of delivering geographic information to a large pool of users. As opposed to desktop GIS where one installs the GIS software in a personal computer, web GIS could be thought of as a distributed information system that provides access to distributed data and other spatial/non-spatial information and also conducting GIS analysis.

Web GIS technology has not only enabled visual interaction with geospatial information possible but also widely accessible from almost any location thus speeding the evaluation process (Alesheikh, *et al.*, 2002).

Various standards have been developed to guide implementation of web GIS based technologies. Open Geospatial Consortium (OGC) which is a non-profit making organization with representatives from the GIS industry develops standards called OGC specifications. Among others, the following standard protocols facilitate serving of geospatial data on the web (OGC, 2021).

i. Web Map Services (WMS)

WMS provides a HTTP interface for requesting geo-referenced map images from one or more distributed geospatial databases. The response to the request is one/more geo-referenced map images that can be displayed in a browser at different zoom levels according to specifications.

ii. Web Feature Service (WFS)

WFS permits users to access and manipulate geospatial information. The data served by WFS is normally in vector format.

2.3. GeoDjango

GeoDjango is a python-based geographic Web framework whose goal is to make it as easy as possible to build robust and secure GIS Web applications and harness the power of geospatial data (Django, 2021). It is built on top of Django framework and extends its core features. Some of the notable features of GeoDjango include:

- Django model fields which map to the database and model OGC geometries and raster data.
- Extensions and plugins to Django's Object Relational Mapper (ORM) for storing, organizing, querying and manipulating geospatial datasets.
- Loosely-coupled, high-level Python interfaces for GIS geometry and raster operations and data manipulation in different formats.
- Editing geometry fields from the administration.

GeoDjango inherits Django's MVC (Model View Controller) framework where the model is used to design the database, View is used for creating templates for visualization and Controller is where the logic happens (request-response cycle between client and server). To facilitate manipulation and the visualization of geographic data, GeoDjango relies on several third-party libraries which include the following:

i. GDAL/OGR

One of the most important ones is GDAL, which is a Geospatial Data Abstraction Library necessary for conducting queries and data manipulation for raster datasets. As a library, it represents a single abstract data model to the calling application for all the supported formats. GDAL enables read and write raster data in different formats while OGR facilitates reading and writing data in different vector formats.

GDAL uses certain datasets from external authorities to show spatial reference information of datasets. The most common one is European Petroleum Survey Group (EPSG).

ii. GEOS

GEOS stands for Geometry Engine - Open Source and is a C++ library adapted from Java Topology Suite (an open-source Java software library that provides an object model for Euclidean planar linear geometry together with a set of fundamental geometric functions). GEOS implements the OpenGIS Simple Features for SQL spatial predicate functions and spatial operators such as Union, Distance, Intersection, Symmetric Difference, Convex Hull, Envelope, Buffer, Simplify, Polygon Assembly, Valid, Area, Length. GeoDjango abstracts these functionalities by implementing a high-level Python wrapper which are bindings to the GEOS library (Django, 2021).

2.4. Parking and Parking System

A parking lot refers to a dedicated area that most times has been paved intended for parking vehicles. Most car lots are marked to indicate the spatial extent of the respective lots. Parking features either on-street, off-street or buildings, which may be private or public.

Parking is very dynamic; a parking lot may be engaged and in the next 10minutes the lot is unoccupied. Locating a parking space especially in highly populated urban areas has become a daily challenge for many drivers as per the IBM global parking survey in 2011 (IBM, 2011). In the year 2010, the IBM survey found that nearly 6 of 10 drivers abandoned their search for parking lot and more than a quarter had an argument with a fellow motorist over a parking lot.

According to this report, drivers who drive around looking for a vacant parking lot cause over 30% of traffic in a city. In Nairobi for instance drivers averaged 31 minutes in their longest search for a parking slot while worldwide the average time was 31 to 40 minutes of driving around while looking for a parking.

To address the challenges inherent in parking spaces searching, various Parking Guidance Information Systems (PGIS) have been developed to provide incremental solutions, but they operate as standalone systems often in the specified areas and therefore disseminate in formation only to few or selected users which may not reach new users. Despite installation of these systems, their usage remains low due to inaccurate or out-of-date information displayed (Yanjie, et al., 2012). The objective of PGIS is to help in regulating parking demand in space and time, improve parking space utilization and reduce road traffic arising from driving around looking for parking (Watene , *et al.*, 2013). A PGIS needs to meet the following requirements (Ting, in Watene , *et al.*, 2013):

i. Collect real-time information about status car parking spaces in terms of utilization and deliver such information to management center in real time for processing.
ii. Imply automated car park management including analysis of data.
iii. Parking spaces information status should be collated and queried in the information management center in an efficient manner.
iv. Use and supply the information of parking location and status of the parking space to the drivers.

The above requirements can only be met if the PGIS has the following four modules; information collection, information processing, information transmission and information dissemination modules.

The information dissemination module includes a variety of publishing tools and platforms such as LED screen, traffic information broadcasting and television, Internet, GIS, and car navigation system (Watene , *et al.*, 2013).

2.5. Technologies Review

Two major technologies have been deployed to enable efficient management of parking spaces. These technologies have had their own share of challenges hence investments in research have been done to gradually address these challenges. The two major technologies are Smart Parking Management Systems and Intelligent Transportation Management Systems. To address the challenges faced by these two technologies a GIS based parking management system has been proposed which in addition to working as a stand-alone, can also be integrated with the two aforementioned technologies.

2.5.1. Smart Parking Management System

This technology employs various technologies notably deployment of wireless parking meters mounted on stable platforms with sensing and communication capabilities allowing drivers to obtain real-time parking information on their destinations and alleviating parking conflicts (Hongwei , 2011).

Many systems are not intelligent enough because they cannot help drivers find a desired parking space in crowded areas, sometimes making the situation worse if wrong information is relayed to the drivers (Rashid, *et al.*, 2012) . Detailed information on parking availability and utilization would allow drivers make better real-time decisions on use of parking lots and roadside parking (Rashid, *et al.*, 2012). To make these systems smarter, Rashid, *et al.*, (2012), proposed a Reservation-based Smart Parking System (RSPS) that not only broadcasts real-time parking price based on the parking availability but also provides reservation service as part of user-targeted service.

The performance of a parking management system can be evaluated based on the following metrics:

i. Walking distance- distance from a driver's selected parking lot to the driver's destination.
ii. Parking revenue-aim of any management system is to maximize revenue.
iii. Service differentiation- a driver wishes to pay as little as possible thus the system should differentiate drivers according to their budget and need which would be accomplished by having time-based parking.
iv. Traffic searching for parking- the system should reduce amount of searching time for parking by ensuring that drivers can book a slot prior to the journey and specify the time the lot would be occupied.

2.5.2. Intelligent Transport Systems (ITS)

ITS aims to provide innovative solutions to the transport sector by application of various technologies such as car navigation, traffic signal control, automatic number recognition and speed cameras. In the car parking industry, ITS objective is to extract and recognize vehicle registration numbers from car images using various machine learning algorithms, process the image data and

utilize the information for parking lot access record (Rashid, *et al.*, 2012). Using this technology, information regarding parking free spaces can be relayed to users in real-time.

ITS would be categorized in the line of image acquisition and processing to determine whether parking lots have objects on them and if there are objects, the image processing is able to judge whether it is a vehicle or not. Collating this data, the number of parking spaces available is relayed to the control center.

The ITS and the smart parking management system may have a drawback called *multiple-user-chasing-single-space* (Rashid, *et al.*, 2012). This is a phenomenon whereby a parking lot has very few parking spaces available in busy hours and more drivers struggle for less parking spaces causing severe congestion.

2.6. **GIS based Parking Management System**

To address the *multiple-user-chasing-single-space* phenomenon a GIS based parking management system may be integrated with the above reviewed technologies by using GIS as a tool to integrate these datasets, perform spatial analysis and visualization of the information.

The data into the GIS can also be generated by the users of the system, a phenomenon known as user-generated content (UGC). UGC is a situation in which content displayed on a system or service is created by the users of that system or service (ESRI, 2021). The system in this case will employ a special type of UGC called Volunteered Geographic Information (VGI) which drivers would provide the information by having the parking lot they have occupied marked as occupied.

The unique power of a GIS-based system in addition to being capital intensive for a start, is preferred due to but not limited to the following capabilities:

i. Analysis- spatial analysis is the core of GIS since it includes all transformations, manipulations and methods applied to geospatial data to add value to, support decision making and reveal patterns in data through modelling. Spatial analysis is built by majorly four spatial relationships; containment, distance, adjacency and selection and statistics. These relationships are actualized by several geometric algorithms such as metric, topological, set-based and triangulation algorithms.
ii. Integration- GIS handles spatial and non-spatial data in a seamless manner and performs analysis and disseminate to concurrent users with different backgrounds.
iii. Data Management- GIS databases are able to manage data in various forms such as vector and raster with their respective spatial information. More often data from the parking lots cannot be handled by the regular relational databases and therefore on would need a database that can store topological information.
iv. Visualization- with the exponential democratization of GIS, communication with the audience or users is no longer a challenge. Users can perform their own local analysis on the client and visualize the results. 3D geovisualization which entails visualizing geographic information in any of the steps of spatial analysis including the time component makes GIS based systems easily understandable to the masses.

The result would be a web interface (map) updating itself in real-time to reflect the parking situation in the area of study. The web map allows anyone to visualize the parking situation before making a trip. Parking lots are displayed on the web map each identified uniquely to enable users

to reserve a lot. Components such as navigation to the parking spaces, facilities around and predicted situation in time t-T can also be integrated in the interface.

Once a driver reserves a parking lot it is marked unavailable for that duration of time until the driver signs out and the parking lot is released. This information would help users decide the best places to park based on the anticipated traffic generated by the parking in addition to the walking time of the user to the destination.

2.7. Case studies

The area of GIS and transportation has risen to prominence over the years especially in the wake of having smart cities. GIS is being used for input, storage, retrieval, management, analysis and output of location-based information.

2.7.1. US case study

Advanced Parking Management Systems (APMS) help drivers find parking slots quickly, therefore reducing frustration and enhancing visitors' experience. APMS come with various types of benefits and as a consequence has been adopted by many communities across the US. For example, in the Baltimore-Washington International (BWI) Airport, the system was found to have increased customer satisfaction and improved traffic flow in the hourly facilities (FHA, 2007). APMS has also been widely used in Europe and Japan with notable applications in festival locations and sports arenas to reserve parking.

In the study, Advanced Parking Management Systems, a cross cutting study, *Taking Stressing out of Parking (2007), the* Federal Highway Administration identified three cases of what comprised the Parking Challenge (FHA, 2007):

i. Public parking is available, but travelers have little or no information on where parking facilities are, hours the facilities operate or the cost of parking in such lots.
ii. Public parking is in short supply and privately owned parking is available with low levels of utilization.
iii. There is a shortage of parking facilities and drivers are unaware of this fact until they arrive at their destination hence resulting to multiple parking.

In all of the above cases, the Federal Highway Commission concluded that travelers lack credible, timely information which causes travelers to make bad decisions and exhibit poor driving habits due to frustration in not locating a parking slot.

Several cities have implemented pre-trip information systems in the form of webpages, which provide a map of where parking facilities are relative to access routes and attractions.

2.7.2. Kenyan case study

This study entailed designing of a parking management system that based on a mobile platform thus allowing the driver to access parking information wherever he/she is. The resultant system allowed drivers to view the real time parking situation of his/her destination and reserve parking. Thirty outdoor parking spaces were used in this study. The parking spaces were assigned unique identifiers. The parking lots were assigned QR codes printed on metallic plates from which drivers

would scan and send information indicating that the lot was occupied. If a parking lot was occupied, it disappeared from the map until it was declared vacant. A booking facility was availed in the system which allowed the driver to send his/her vehicle registration number and receive SMS with the respective details. A message would also be sent to the parking attendant alerting that a particular parking lot has been reserved.

3. MATERIALS AND METHODS

3.1. The study area

The study area for the project was a selected part of Upper Hill area located approximately 4 kilometers by road west of Nairobi Central Business District. The area lies in zone 37M of the Universal Transverse Mercator between Northings 9855870-9857580 and Eastings 257840-256860 metres (Figure 1).

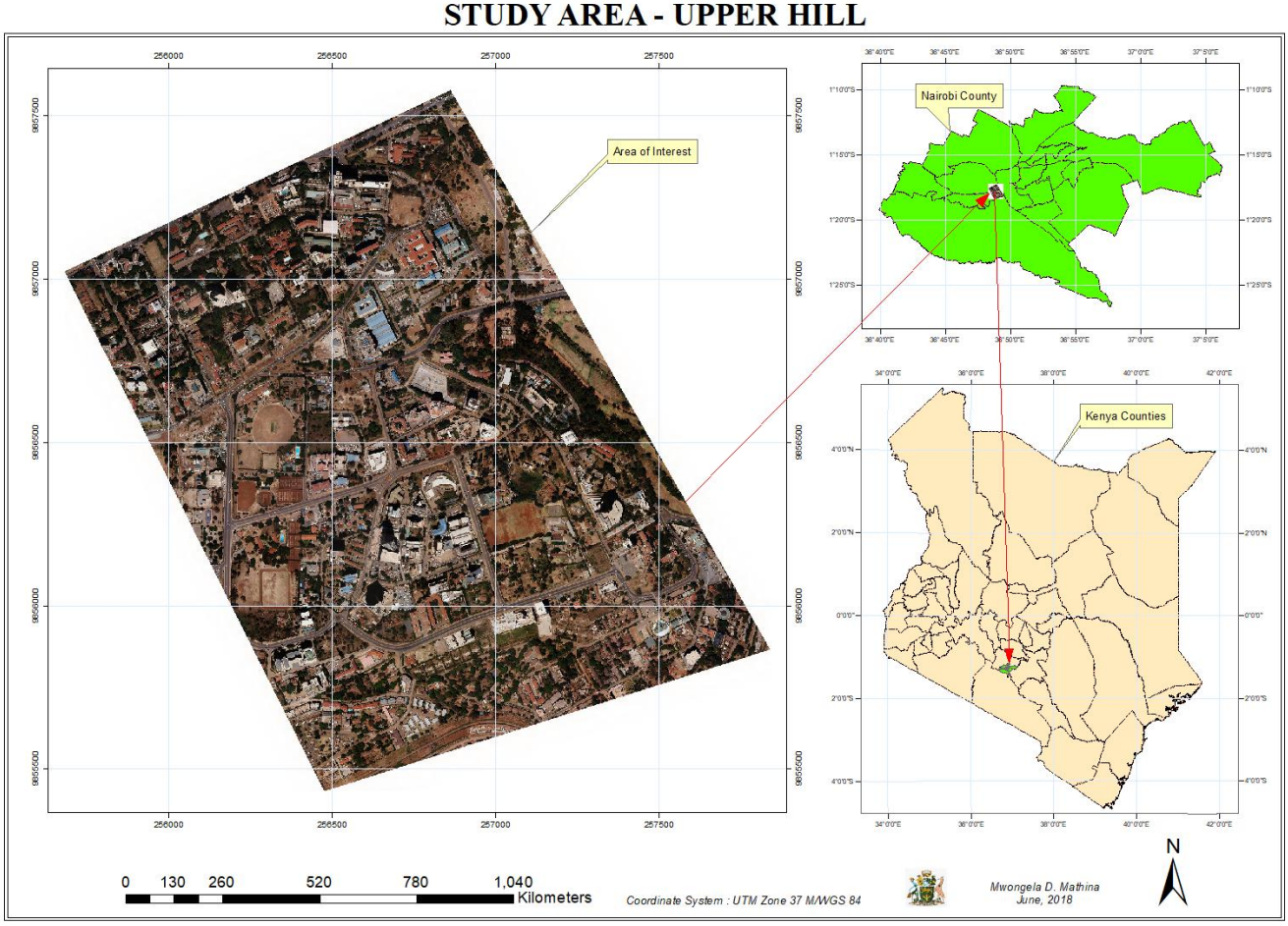

Figure 1: The study area

Recently Upper Hill has become amongst the areas businesses are shifting their operations to and hence increase in business activities which leads to not only demand for office space but also demand for car parking facilities. This increase in demand for the limited resources such as car park spaces needs be met by innovative ways that ensure proper utilization of the parking spaces but also increased efficiency of how the lots are managed. This reason makes Upper Hill an ideal area of study to test a system that ensures efficient management of parking lots.

3.2. Methodology for the Study

The following flow chart (Figure 2) guided the development of the system.

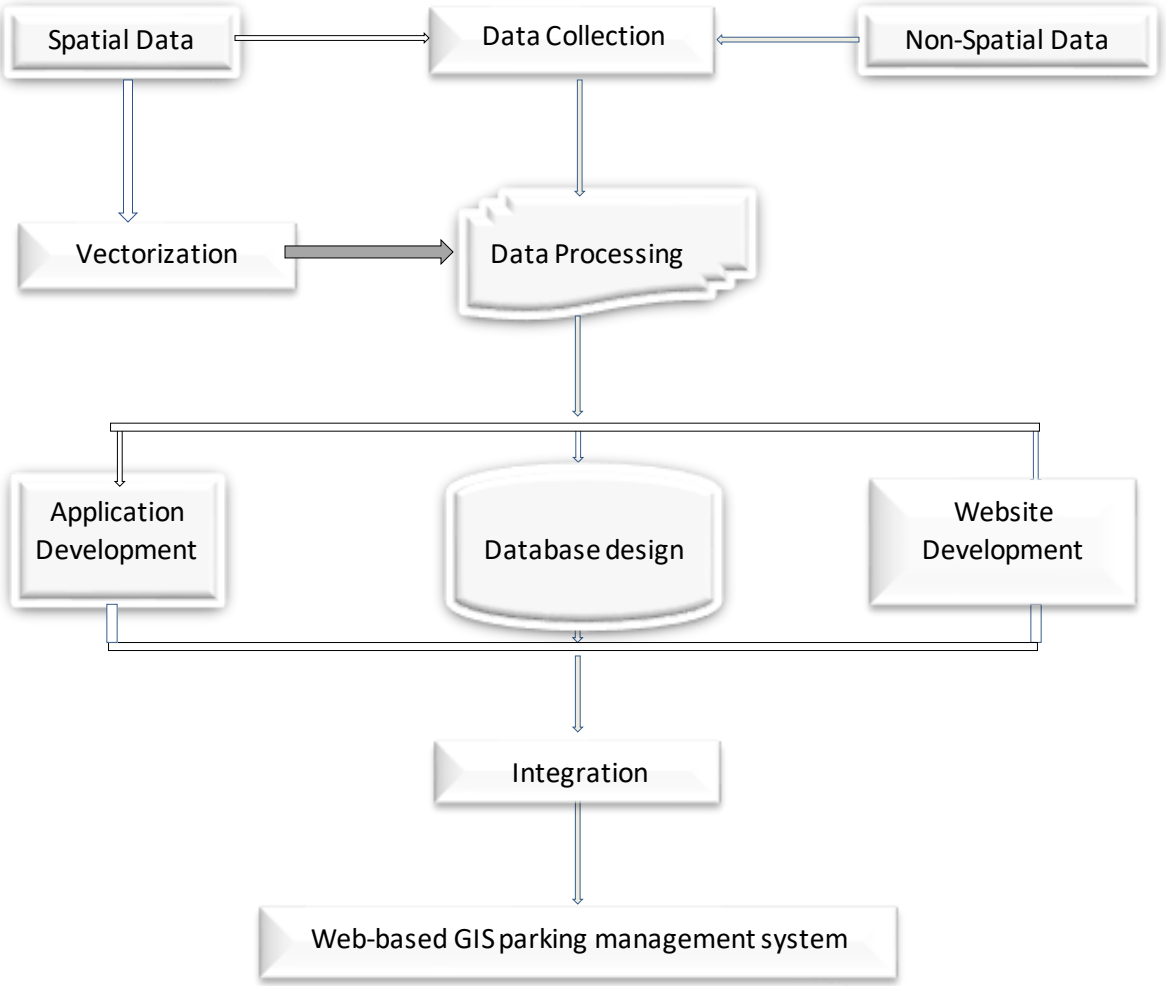

Figure 2: Study workflow

3.1. Software, Hardware and Data Sources

The hardware used was mainly a personal computer while open-source software was used except for the google maps basemap. Open-source software which is maintained by a large community of developers offers the advantage of being highly customizable, constant improvements and being battle tested.

A summary of the software and tools that were used in the study is shown in Table 3.

Table 3:Used Software and Tools

Item	Category	Source	Remarks
Orthophotograph	Data	Ramani Geosystems	Free
GeoDjango	Software/Framework	https://www.djangoproject.com/	Open source
Django REST Framework	Software	https://www.django-rest-framework.org/	Open source
React	Software/Library	https://reactjs.org/	Open source
pgAdmin4	Software	https://www.pgadmin.org/download/	Open source
QGIS	Software	https://qgis.org/en/site/forusers/download.html	Open source
Personal Computer	Hardware	Self	Initial cost

3.2. Data Collection

The data collected was mainly secondary data. The dataset to be extracted was the parking lots which were obtained from the aerial photograph.

3.3. Creation of a Geodatabase

A geodatabase is an object-relational database that enables storage of spatial and non-spatial data in a single database. Organization of the datasets into a geodatabase ensures integrity of the dataset is kept. A file geodatabase was created in Arc-catalog and feature datasets such as parking lots and parking lots attributes created in the geodatabase. Data was added to the feature classes with indexes for the appropriate fields to improve query performance in data access.

3.4. Parking lots digitization

The parking lots were extracted from the orthophotograph by on screen digitizing. High resolution of the photography enabled very fine details of the parking lots to be derived for instance whether the parking lots are reserved for disabled among other key attributes. Unique numbers were assigned to the vectorized parking slots to identify the characteristics of each slot uniquely. Figure 3 shows the extracted parking lots.

Figure 3: Parking lot extraction

3.5. Application Development

The architecture of the application followed the MVC pattern and a REST API that provide JSON data to the client side as shown in Figure 4. For the system to be available online, the Django module was used. To make the application spatial, the GeoDjango module built in Django extends the Django applications to make them harness the power of spatially enabled data was used.

GeoDjango is supported by GDAL/OGR which is a third-party library and therefore was one of the requirements to be installed.

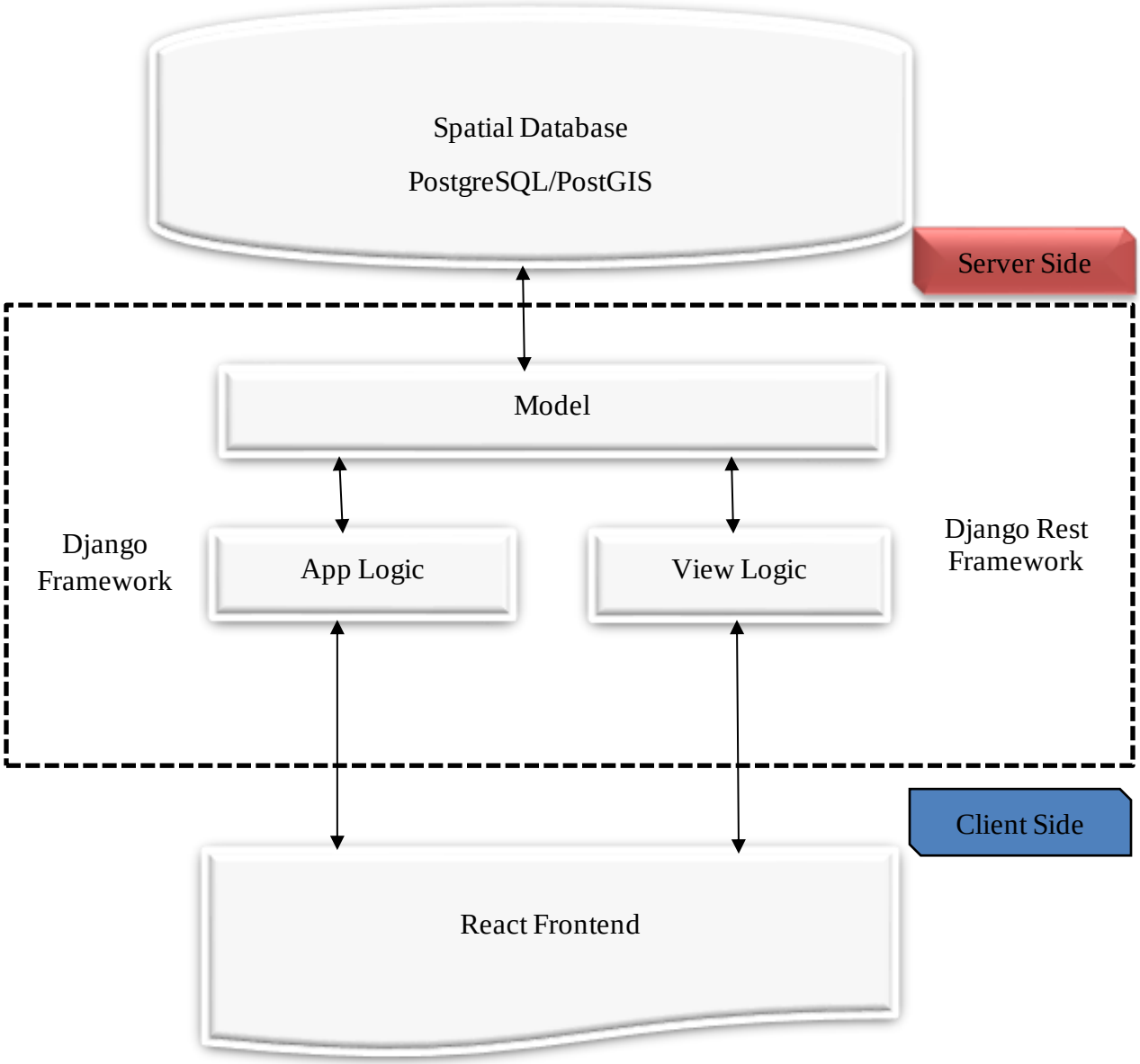

Figure 4: Architecture of the parking system

3.1. Database Design

3.1.1. Database schema

Django allows creation of a database schema without having to write the SQL scripts, instead this is highly abstracted. This is done by creating models in Django in a models.py file that will contain all the fields of the entity in question. A Django model is a description of the data in a database, represented as Python code. It's the data layout–the equivalent of SQL CREATE TABLE statements except, it's in Python instead of SQL.

The model executes the SQL scripts and returns python data structures representing rows in the database tables thus the database schema. In this case, PostgreSQL database which has a more developed PostGIS extension that handles spatial data was used as the database. The schema created is then ready to be populated by the prepared datasets from a GIS environment-often shapefiles. The **ogrinspect** management command which is run on the command window inspects the shapefiles from the GIS environment and generates model fields which match the fields created in the feature class. This command was used to generate the matching fields for the datasets.

Three tables were generated, the Lots table represents the parking lots in polygon format and were used to link the details of the users and also the status of the lot to show whether it is occupied or not. The ParkingLots table shows the parking lots in line format to enable the drivers visualize how the parking lots look like on the ground. The Attributes table shows the information of a group of parking lots such as whether it is private, the type of vehicle to be parked and the road the lots are located along. The resulting tables were generated as shown in Figure 5 and Figure 6.

```
from django.contrib.gis.db import models
from django.contrib.auth import get_user_model
class Lots(models.Model):
    driver = models.ForeignKey(get_user_model(), null=True, on_delete=models.CASCADE, relate
    geom = models.MultiPolygonField(srid=4326)
    occupied = models.BooleanField(default=False)
    booked = models.BooleanField(default=False)
    regNo = models.CharField(max_length=8, blank=True,null=True)
    phoneNumber = models.CharField(max_length=10, blank=False, null=True)
    when = models.DateTimeField(null=True)
    bookedTime = models.CharField(max_length=3, null=True)

    def __str__(self):
        return str(self.regNo)

    class Meta:
        verbose_name_plural = "Lots"
```

Figure 5: Python code generating the lots database schema

```
class ParkingLots(models.Model):
    geom = models.MultiLineStringField(srid=4326)

class Attributes(models.Model):
    road = models.CharField(max_length=50)
    vehicle = models.CharField(max_length=50)
    assistant = models.FloatField()
    type = models.CharField(max_length=50)
    geom = models.MultiPointField(srid=4326)
```

Figure 6:Python code generating the parking lots and attributes database schema

3.1.2. Database Population

A database schema is an empty representation of the fields as specified in the model. For the database to be useful, it needs to be populated with datasets. The database used in this study is the PostgreSQL database which provides a lot of functionality in manipulating geospatial data enabled by extensions such as PostGIS. The PostgreSQL database is connected to the web application by setting the User, Name of the database, Port, Password and Host. The database can be created via the terminal or command window or in the pgAdmin user interface of the PostgreSQL database. In this case a python script that automatically maps the shapefile fields to the Postgre SQL database was used. To populate the database, the LayerMapping data import utility was used. The LayerMapping class provides a way to map the contents of vector spatial data files (for example, shapefiles) into GeoDjango models. This utility provided an easy way to avoid repetition when populating the database as shown in Figure 7.

```
lot2_shp = Path(__file__).resolve().parent /'data'/'data' / 'lot2.shp'

attributes2_shp = Path(__file__).resolve().parent /'data'/'data' / 'attributes2.sh


lot_1_shp = Path(__file__).resolve().parent /'data'/'data' / 'lot_1.shp'

def run(verbose=True):
    lm = LayerMapping(ParkingLots, str(lot2_shp), parkinglot_mapping ,transform=Fa
    lm.save(strict=True, verbose=verbose)

def run_attributes(verbose=True):
    lm = LayerMapping(Attributes, str(attributes2_shp), attributes_mapping ,transf
    lm.save(strict=True, verbose=verbose)

def run_lots(verbose=True):
    lm = LayerMapping(Lots, str(lot_1_shp), lots_mapping ,transform=False )
    lm.save(strict=True, verbose=verbose)
```

Figure 7: A Python script that loads the parking lots to the PostgreSQL database

3.2. Website Development

A website was created for the frontend and customization of the backend was done. To create the front-end HTML, leaflet (JavaScript) library, google maps and CSS styles were used. The GeoJSON data was served from the database via URL endpoint running on the Django framework webserver. The axios library which is third part JavaScript library was used to fetch the serialized data from the server and send it to the client. On the client side the React Library which helps build user interfaces with code optimization and fast resource loading was used. This ensured that the time taken for the user to load data from the database and display it for decision making is greatly reduced.

3.3. WebGIS Based Vehicle Parking Management System

After integrating the server side and the client side, the result was a vehicle management system. The components of this system include the database, the application and the website. The database is available for updating by the administrators who have privileges to update or delete information contained in the system. The database was hosted on the Amazon Web Services (AWS) which is a cloud platform that provides hosting services. The system was hosted on Heroku – a cloud platform that makes it easy for geospatial based applications to be easily brought online.

4. RESULTS AND DISCUSSIONS

4.1. Demonstration of the database contents

The database that powers the vehicle parking management system is based on the PostgreSQL database. The contents of such a database can be viewed on the administrator's end using the pgAdmin platform which presents a graphical view of the database. The table created in the database in this case for the parking lots data (polygons) is shown in Figure 8.

```
SELECT * FROM public.parkinglots_lots
ORDER BY id ASC
```

Output | Explain | Messages | Notifications

id [PK] integer	geom geometry	booked boolean	occupied boolean	regNo character varying (8)	phoneNumber character varying (10)	driver_id integer
1	0106000020E61000...	false	false	[null]	[null]	[null]
2	0106000020E61000...	false	false	[null]	[null]	[null]
3	0106000020E61000...	false	false	[null]	[null]	[null]
4	0106000020E61000...	false	false	[null]	[null]	[null]
5	0106000020E61000...	false	false	[null]	[null]	[null]
6	0106000020E61000...	false	false	[null]	[null]	[null]
7	0106000020E61000...	false	false	[null]	[null]	[null]
8	0106000020E61000...	false	false	[null]	[null]	[null]
9	0106000020E61000...	false	false	[null]	[null]	[null]
10	0106000020E61000...	false	false	[null]	[null]	[null]
11	0106000020E61000...	false	false	[null]	[null]	[null]
12	0106000020E61000...	false	false	[null]	[null]	[null]

Figure 8: Parking lots view from the database

Each parking lot is given a primary key which increases the querying efficiency of the datasets. Using the primary key, records can be deleted or updated without having to carry out database transactions on the whole database to locate a record.

4.2. Visualization of the parking lots

The parking lots are displayed on a web map whose base map is the google maps due to its user friendliness and many capabilities. Users can pan to any area of interest on the map to view the parking lots available in that area. Figure 9 shows part of the parking lots in the study area. A collection of parking lots is denoted by a parking symbol.

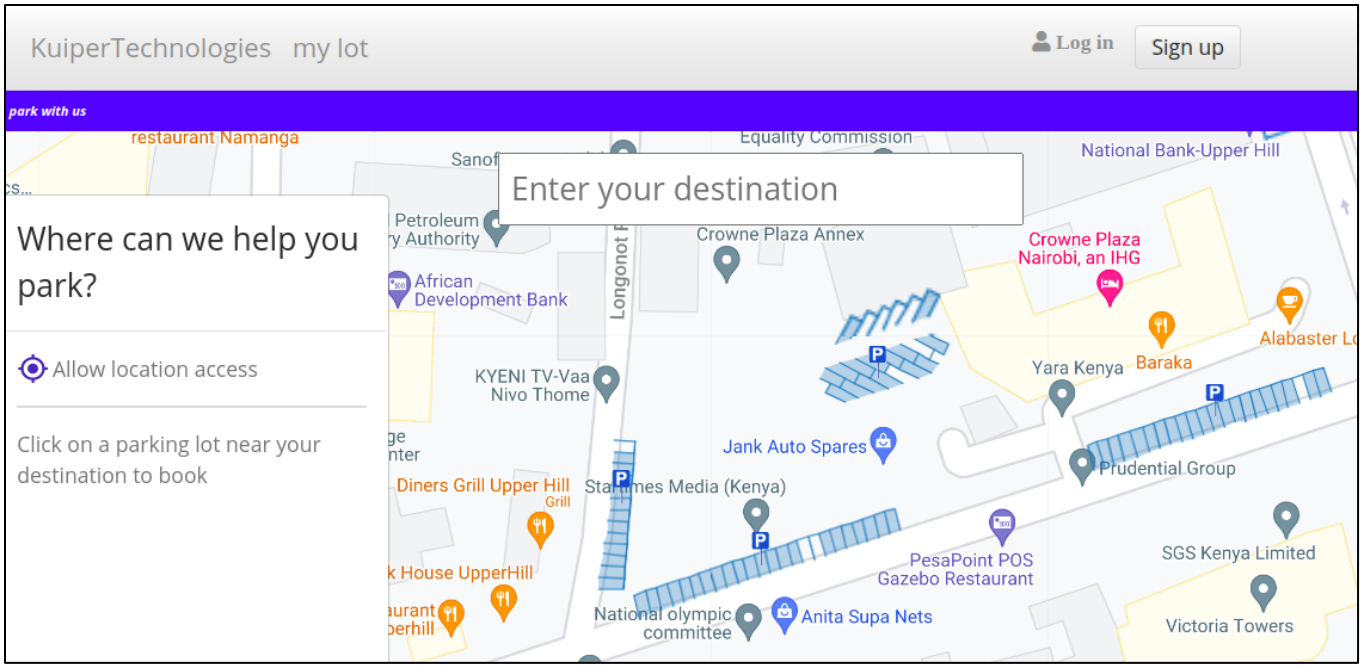

Figure 9: Part of the parking lots in the study area

4.3. System search functionality

The vehicle management system is made in such a way to make it as easy as possible for the user to navigate to any of the destination he or she wishes to explore the parking situation before making a parking decision. Using the search functionality, a user enters any destination and the a pplication zooms to that particular location with the loaded parking lots in that area. Before searching his or her destination the user can click the 'allow location access' button that zooms to the current location the user is at as shown in Figure 10.

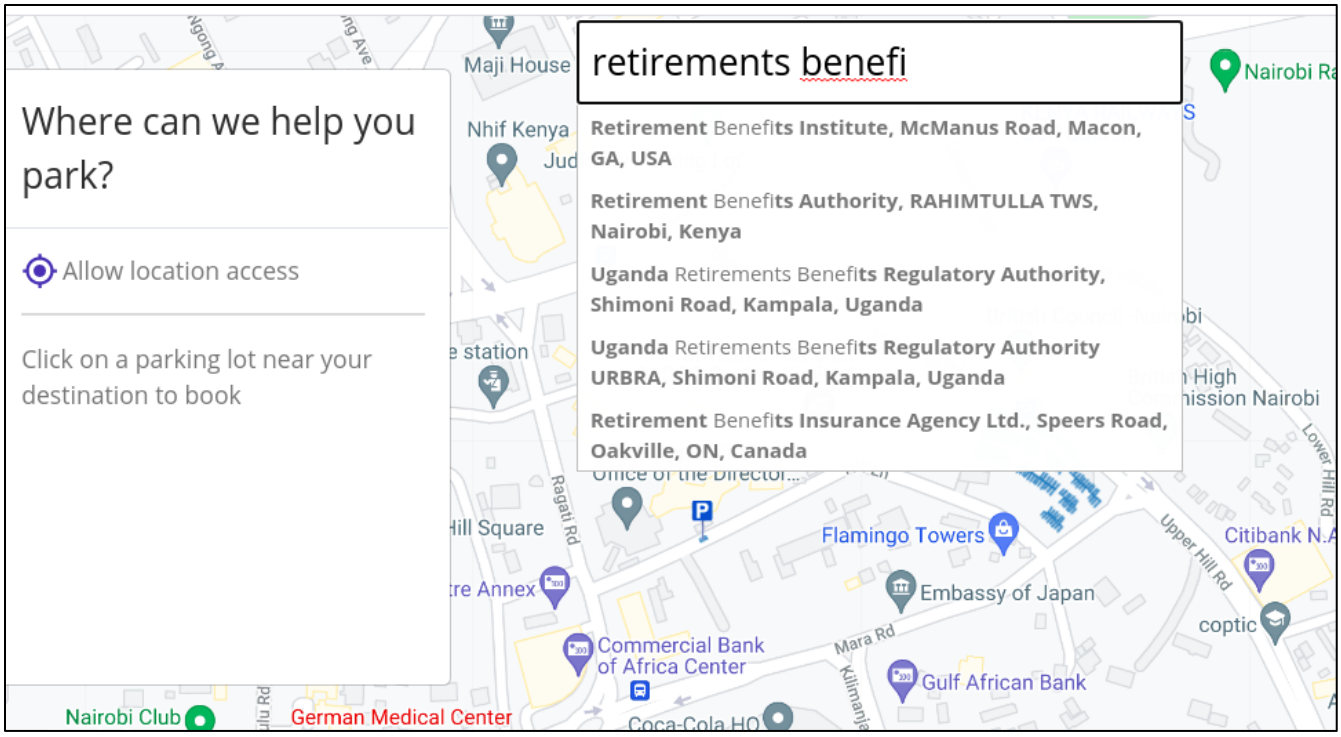

Figure 10:Search functionality of the system

4.4. Real Time parking booking

The parking system offers users from any part of the country real time streaming of the obtaining parking situation. Figure 11 shows a user in the process of booking a lot. When a user clicks on a parking lot, the parking lot is identified uniquely from which the user can click the book button to be redirected to the booking page for completion of the process which includes entry of the user details as shown in Figure 12.

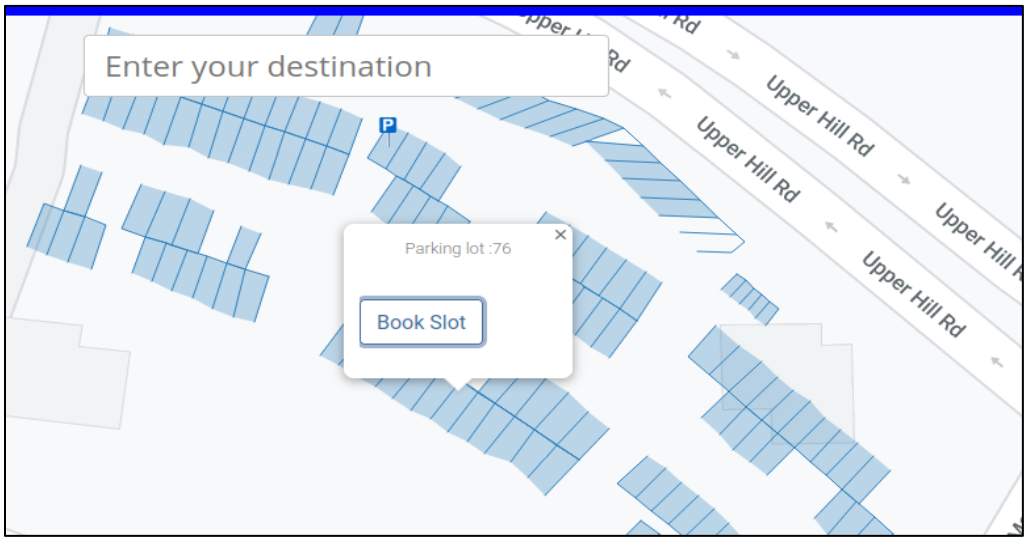

Figure 11: Parking lot booking process

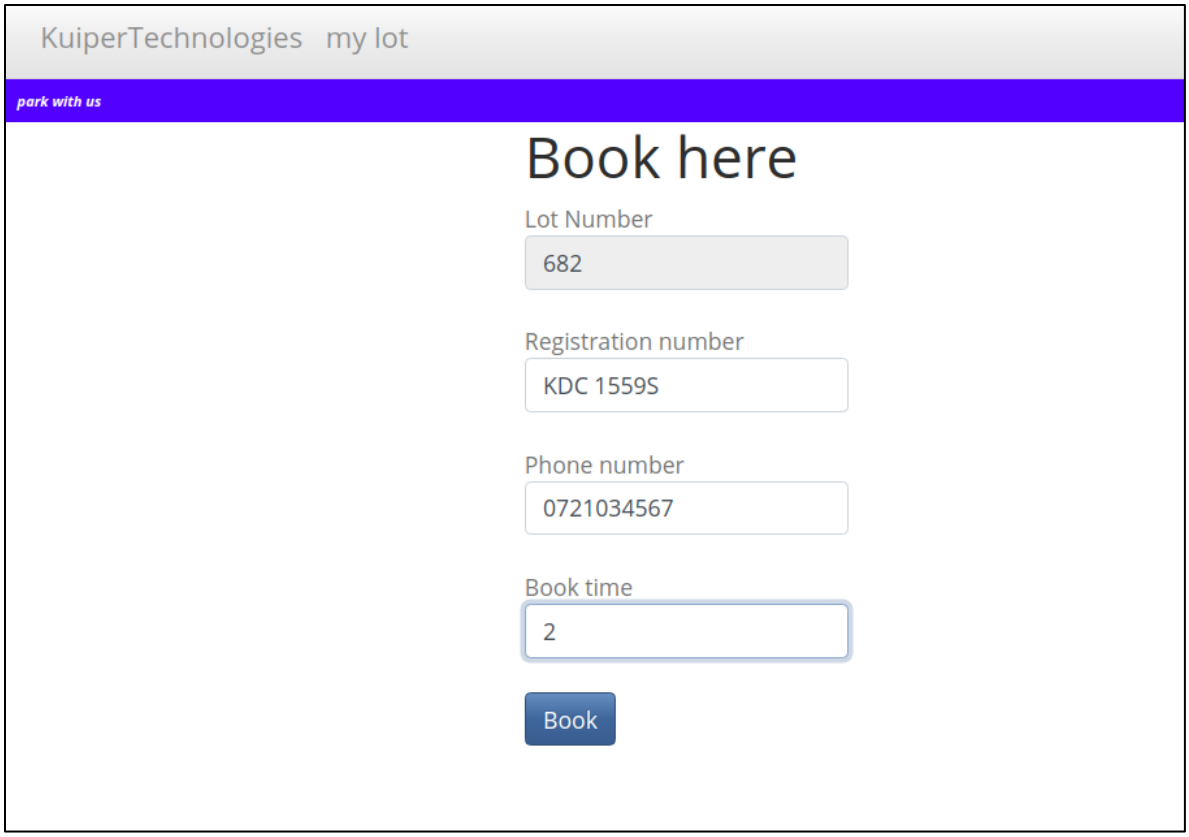

Figure 12: Booking process completion

Once the user successfully books a lot, he/she is redirected to view the booked lot on a map in context with the other lots as shown in Figure 13.

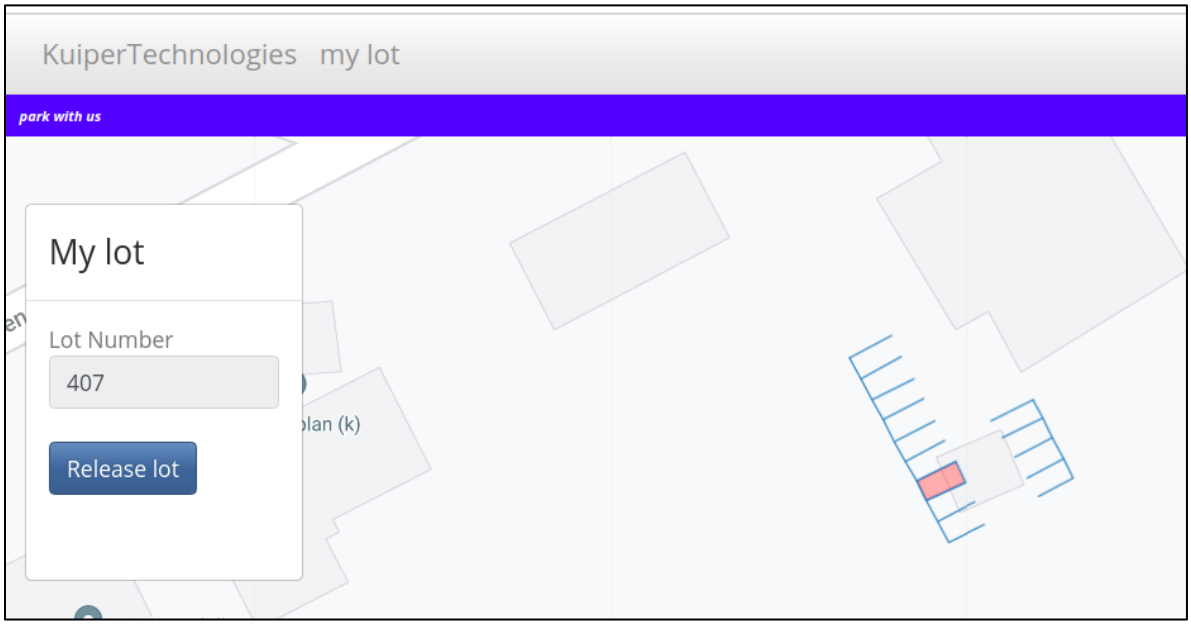

Figure 13: The booked lot

The car parking management system is set up such that a user can only book or occupy a single parking lot at a time. To enforce this, a driver trying to book an additional lot while he/she has a lot in the profile, that request would be rejected as shown in Figure 14.

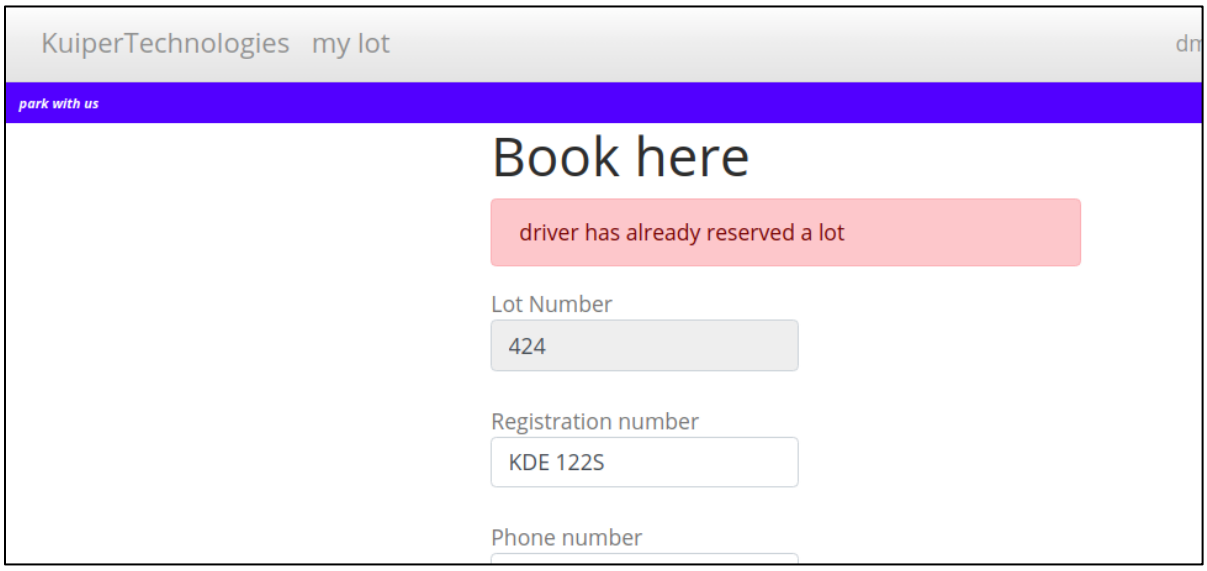

Figure 14: A user can only book a single lot

The system is designed to automatically release the parking lots from the users once the time the users have indicated in their booking details expires. This functionality is also delegated to the user if he/she were to spend less time on the parking lot or abandon the parking lot all together.

4.5. Visualization of occupied parking lots

The parking lots occupied are displayed on the map with a red color style for users to distinguish such lots as shown in Figure 15. As the user decides on where to park, he or she can make the decision based on utilization of the parking spaces in the area of interest.

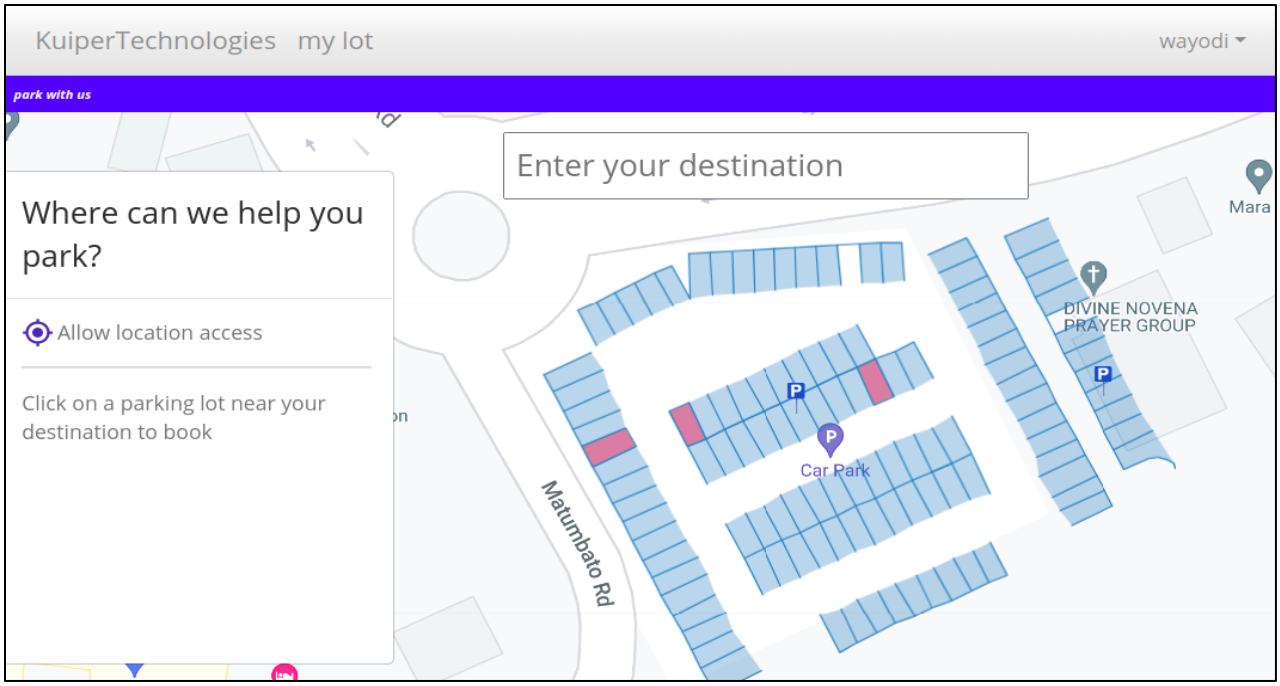

Figure 15: Part of the booked lots in red color

4.6. Other functionalities

The system offers other functionalities such as rejection of requests trying to book already an occupied lot. All users that are not signed in can view all the parking lots but to book one, a user has to register or log in to book a lot. The system restricts users from viewing details of the users who occupy the lots at any point in time.

4.7. Administrators View

The administrator has added privileges such as viewing the details of the users occupying the parking lots by querying the database based on the occupied or booked attribute of the respective table as shown in Figure 16. The administrator has rights to release lots from users and also delete those lots from the system as shown in Figure 17.

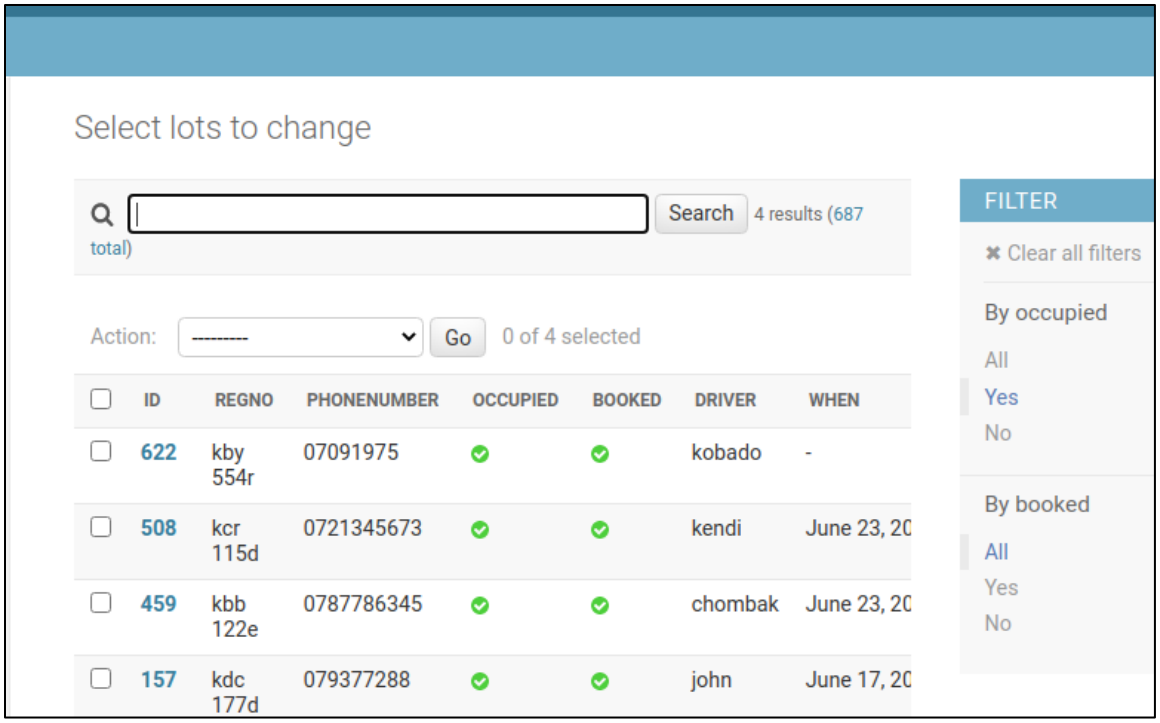

Figure 16: Administrator's view of occupied lots

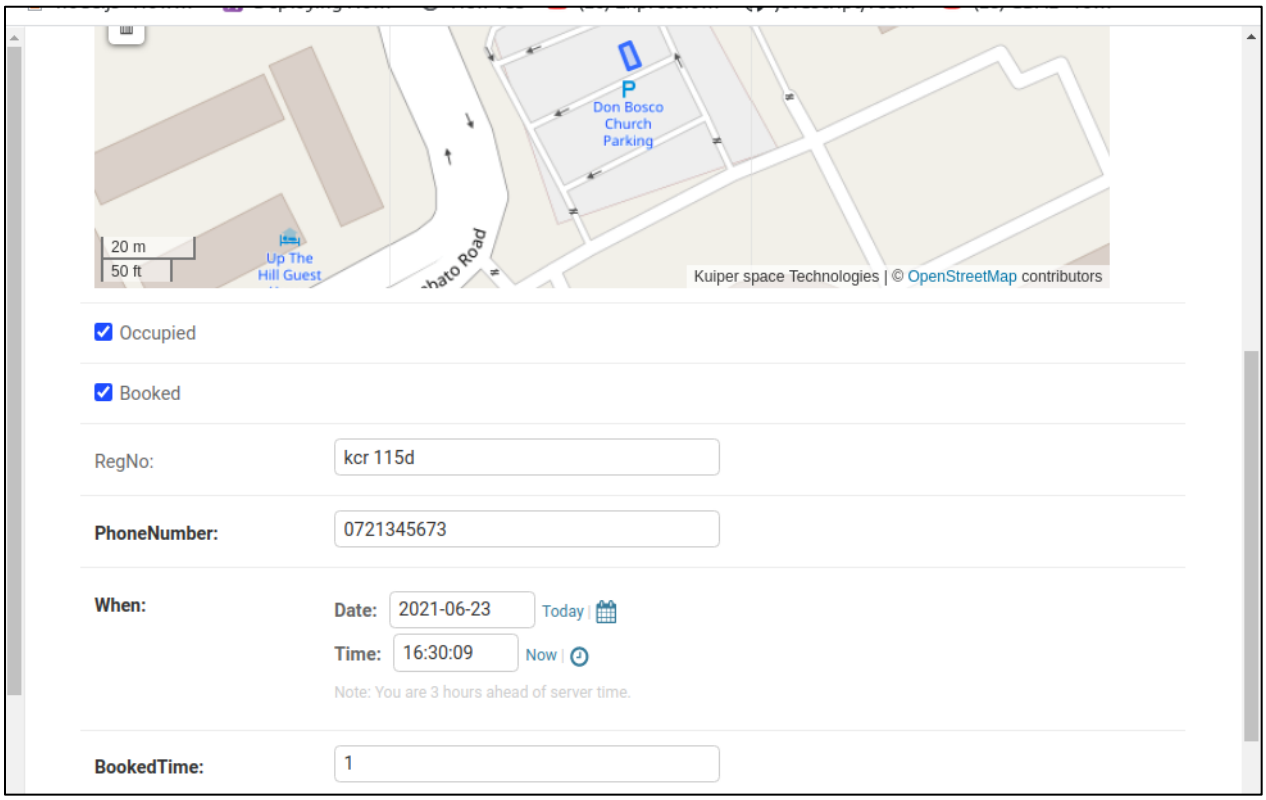

Figure 17: Administrator's view of individual lot details

4.8. Discussion of Results

The results of various capabilities of the system aimed at achieving the objectives of the study are presented in regard to the problems inherent in the current parking management systems. The study aimed to develop and demonstrate an operational webGIS based vehicle management system with real time capabilities and intuitive user visualization experience.

In the current parking management systems drivers are not aware of how the parking situation looks like until they reach their destination, the developed system enables users to have a global view of how the parking situation looks like and make decisions on the most appropriate locations to park their vehicles. In addition, traditional parking management systems focus more on the availability of parking lots in a group of parking lots. This system goes beyond the generalized view of looking at parking lots and focuses on the individual parking lot by focusing on its spatial information in as far as where it is located within the group of the parking lots. This kind of information enables the driver choose the most appropriate parking lot say at the extreme end for easier parking.

One of the challenges of parking in town is users having their cars towed for what is considered as illegal parking. The designed system solves the problem by ensuring that vehicles are only parked at designated lots available for everybody to view. Moreover, in case of any emergency the user's details are available in the system hence swift action.

The unique capability that GIS systems bring on board is visualization; this functionality offers the users unique bird's eye view of the whole city and the parking lots available and the percentage of utilization thus their decisions are backed by data.

The users can access the website of this system at the location https://kuiper-demo2.herokuapp.com.

5. CONCLUSIONS AND RECOMMENDATIONS

5.1. Conclusions

GIS based car parking management systems were seen as better solution in helping drivers locate parking lots conveniently. Consequently, the objectives of the study were achieved as a system that enables real time querying and visualization capabilities was developed. A spatial geodatabase was successfully designed and implemented as demonstrated by spatial queries which could be performed from the web application. The following conclusions were made:

- The developed parking system offers additional advantages to drivers especially foreigners by helping them not only know the parking lots that are vacant but also be guided by other information such as road names where the lots are located and also facilities near the parking lots such as restrooms or parks.
- The developed system can exploit the capabilities of the google maps dataset and thus provide additional information such as driving directions, street view and traffic status on the road.
- Drivers do not have to find themselves at a situation that demands them to stop and look around for parking, the booking of the parking lots can be done while the drivers are on the move depending on changes in traffic. This helps eliminate traffic snarl ups caused by drivers looking for parking.
- The designed system acts as a market place where parking lots belonging to different stakeholders can be hosted and made available for booking to all the users. Integrating with e-payment services the lots owners can introduce time-based parking services which will be attractive to users.

5.2. Recommendations

In lieu of the results obtained in this study, it is recommended that:

1. The developed system should be tested first before attempts to implement it. If the system is tested successfully, the researcher recommends that the system be adopted by all major town's authorities in the Country in managing parking which will not only increase the revenues generated therein but also help users who are new to such places find parking with ease and satisfaction.
2. To improve user experience, cameras should be used to identify parking lots which are not occupied without the user being bothered to update the system. This would also be good for the security of the user so that his/her whereabouts are not easily tracked. This can be achieved by employing machine learning techniques and Artificial Intelligence (AI) systems.
3. To avoid users having to check the obtaining parking situation to book the lots, the researcher recommends that a car dashboard-based system/application be developed to notify users once they are near unoccupied lots to avoid accidents which may be caused by users having to constantly check their phones.
4. The backbone of this kind of system is the spatial datasets which are the parking lots. The researcher recommends that, parking management being a devolved function, the County

governments should undertake mapping and create a database of all the parking lots and provide reliable internet access in the town centres as a pre-requisite to implementing this system.

6. References

Abdi , D., 2019. *Looking beyond on-street parking.* [Online]
Available at: https://www.standardmedia.co.ke/opinion/article/2001327927/looking-beyond-on-street-parking
[Accessed 6 April 2021].

Alesheikh, A., Helali, H. & Behroz, H., 2002. *Web GIS: Technologies and Its Applications.* Ottawa, s.n.

Django, 2021. *geodjango.* [Online]
Available at: https://docs.djangoproject.com/en/3.1/ref/contrib/gis/geos/
[Accessed 3 April 2021].

ESRI, 2013. *Implementing Web GIS.* [Online]
Available at: https://www.esri.com/about/newsroom/arcnews/implementing-web-gis/
[Accessed 10 March 2021].

ESRI, 2021. *The Role of Volunteered Geographic Information in a Postmodern GIS World.* [Online]
Available at: https://www.esri.com/news/arcuser/0410/vgi.html
[Accessed 16 March 2021].

FHA, 2007. *Advanced Parking Management Systems: A Cross-Cutting Study. Taking the Stress Out of Parking,* 1200 New Jersey Avenue: s.n.

Goodchild, M., 2007. Citizens as sensors: the world of volunteered geography. *GeoJournal volume,* Volume 69, pp. 211-221.

Hongwei , W., 2011. *A Reservation-based Smart Parking System.* Lincoln, Nebraska, s.n.

IBM, 2011. *IBM.* [Online]
Available at: https://www-03.ibm.com/press/us/en/pressrelease/35515.wss
[Accessed 5 April 2021].

IPI, 2018. *2018 Emerging Trends in Parking,* Alexandria, Virginia : s.n.

JICA, 2006. *The study on master plan for urban transport in Nairobi metropolitan area in Republic of Kenya,* Nairobi: s.n.

Katahira Engineers International, RECS International Inc, 2005. *The Study on master plan for urban transport in Nairobi metropolitan area,* Republic of Kenya: s.n.

Nairobi City County, 2014. *The Project on Integrated Urban Development Master Plan for the City of Nairobi in the Republic of Kenya,* Nairobi: s.n.

OGC, 2021. *OGC Standards.* [Online]
Available at: https://www.ogc.org/docs/is
[Accessed 5 April 2021].

Rashid, M., Musa A., Atahur Rahaman, Farahana N., 2012. Automatic Parking Management System and Parking Fee Collection Based on Number Plate Recognition. *International Journal of Machine Learning and Computing,* 2(2), pp. 1-6.

Watene , G., Gouglas , M. & Charles , N., 2013. A GIS Based Parking Management and Dissemination System. *International Journal of Science and Research (IJSR), India Online ISSN: 2319-7064 ,* 2(7), pp. 1-8.

Yanjie, J., Amy, G. & Philip T. , 2012. Design the parking guidance information for the drivers.

Zhong-Ren, P. & Ming-Hsiang , T., 2003. *Internet GIS-Distributed Geographic Information Services for the Internet and Wireless Networks.* 1st ed. New Jersey: John Wiley and Sons.

APPENDIX: CODE

Sample code showing the models which map to the database and how the application interacts with the database to generate responses and send to the user.

The entire code can be found at this location: https://github.com/Dominic-Mathina/kuiper

```python
from .serializers import ParkingLotsSerializer, AttributesSerializer, LotsSerializer
from rest_framework.generics import ListAPIView
from rest_framework.views import APIView
from parkinglots.models import ParkingLots, Attributes,Lots
from rest_framework.permissions import AllowAny, IsAuthenticated
from rest_framework.response import Response
from django.shortcuts import get_object_or_404
from django.core.exceptions import PermissionDenied
from django.utils.translation import gettext_lazy as _
from rest_framework import serializers, exceptions
from users.models import User
import datetime

class ParkingLotsView(ListAPIView):
    queryset = ParkingLots.objects.all()
    serializer_class = ParkingLotsSerializer
    permission_classes = [AllowAny]


class AttributesListView(ListAPIView):
    queryset = Attributes.objects.all()
    serializer_class = AttributesSerializer
    permission_classes = [AllowAny]

class LotsListView(ListAPIView):
    queryset = Lots.objects.all()
    serializer_class = LotsSerializer
    permission_classes = [AllowAny]


class MyLotAPIView(APIView):
    permission_classes = [IsAuthenticated]

    def get(self,request,*args, **kwargs):
        queryset = Lots.objects.filter(driver=self.request.user)
        # print(queryset.values_list('id')[0][0])
        lot = queryset.values_list('id').first()
        if not lot:
            lot_msg = _("you have not booked a lot")
            raise exceptions.ValidationError(lot_msg)
            # return Response('you do not have any lot')
        serializer = LotsSerializer(queryset, many=True)
        return Response(serializer.data, status=200)


class BookLotAPIView(APIView):
    permission_classes = [IsAuthenticated]

    def post(self, request, *args, **kwargs):

        lots = Lots.objects.filter(booked=True)
        if lots.filter(driver=self.request.user).exists():
            lot_msg = _('driver has already reserved a lot')
            raise exceptions.ValidationError(lot_msg)
        lotId = self.request.data.get("id")
        regNo = self.request.data.get("number")
        phone = self.request.data.get("phone")
        bookedTime = self.request.data.get("bookTime")
        driver = self.request.user
        lot = get_object_or_404(Lots, id=lotId)
        if lot.booked and lot.occupied != False:
            book_msg = _('parking lot already booked')
            raise exceptions.ValidationError(book_msg)
        if regNo and phone != "":
            lot.booked = True
```

```
            lot.occupied = True
            lot.regNo = regNo
            lot.phoneNumber = phone
            lot.driver = driver
            lot.when = datetime.datetime.now()
            lot.bookedTime = bookedTime

            lot.save()
            serializer = LotsSerializer(instance=lot)
            return Response(serializer.data,status=200)


        else:
            msg = _('All fields must be filled')
            raise exceptions.ValidationError(msg)

class BookedLotsAPIView(APIView):
    permission_classes = [AllowAny]

    def get(self,request, *args, **kwargs):
        queryset = Lots.objects.filter(booked=True)
        serializer = LotsSerializer(queryset, many=True)
        return Response(serializer.data, status=200)


class ReleaseLotAPIView(APIView):
    permission_classes = [IsAuthenticated]

    def post(self, request, *args, **kwargs):
        lotId = self.request.data

        lot = get_object_or_404(Lots, id=lotId)

        lot.booked = False
        lot.occupied = False
        lot.regNo = ""
        lot.phoneNumber = ""
        lot.driver = None
        lot.save()
        return Response('lot successfully released')
```

```python
from django.contrib.gis.db import models
from django.contrib.auth import get_user_model
class Lots(models.Model):
    driver = models.ForeignKey(get_user_model(), null=True, on_delete=models.CASCADE, related_name='mylot')
    geom = models.MultiPolygonField(srid=4326)
    occupied = models.BooleanField(default=False)
    booked = models.BooleanField(default=False)
    regNo = models.CharField(max_length=8, blank=True,null=True)
    phoneNumber = models.CharField(max_length=10, blank=False, null=True)
    when = models.DateTimeField(null=True)
    bookedTime = models.CharField(max_length=3, null=True)

    def __str__(self):
        return str(self.regNo)

    class Meta:
        verbose_name_plural = "Lots"

class ParkingLots(models.Model):
    geom = models.MultiLineStringField(srid=4326)

class Attributes(models.Model):
    road = models.CharField(max_length=50)
    vehicle = models.CharField(max_length=50)
    assistant = models.FloatField()
    type = models.CharField(max_length=50)
    geom = models.MultiPointField(srid=4326)
    occupied = models.BooleanField(default=False)
    booked = models.BooleanField(default=False)
    regNo = models.CharField(max_length=8, blank=True,null=True)
```