# Background and related work

In this section, we introduce the parking problem background and some related existing solutions that aim to organize the parking spaces

## Background

No one can deny that parking related problems is a major concern in most of the modern cities worldwide, especially so in Qatar.

Finding a parking spot may seem like a superficial problem, or an individual problem, but it is in fact a problem that causes major issues in our day to day life. In metropolitan areas worldwide, having a parking area is increasingly important and in order to accommodate the growing demand for parking spaces a significant amount of land and buildings are set aside. For instance, in Australia, parking areas hold great importance especially in Brisbane, Sydney and Melbourne where they have a huge number of parking spaces ranging from 25,633 to 41,687 [8]. Furthermore, estimates show that around 30,000 square kilometers of land in Europe and 27,000 square kilometers in the US are devoted to parking spaces [9].

In addition, parking can be very time consuming and costly. This comes as no surprise, “The U.S. economy bears the brunt of parking pain as 40% of drivers say they have avoided driving to shops due to parking challenges.” [10]. According to a study from car service company Inrix, Americans spend an average of 17 hours per year searching for a vacant parking spot, this leads to a loss of $345 per driver in wasted time and fuel. Similarly, drivers in UK spend an average of 44 hours a year looking for an empty parking, with an estimated total loss of £733.

Contrary to the popular belief that generous parking allocations benefits users, the opposite can happen when there’s an exaggeration in parking allocations. A big parking lot prolongs transportation time and is a waste of useable land. These effects -along with recent land-use, socioeconomic and technological trends- are prompting towns to begin asking some important questions about how to solve the parking problem in a smart and cost-effective way.

For this reason, there have been many attempts to provide smart parking systems as a solution, with each having its own different approach as shown in the Related Work section. The used components in the related works varies and can include Ultrasonic sensors, Infrared sensors, and cameras for vehicle detection. The components above help in providing services to the user such as reserving or checking the availability of a parking spot.

## Related work

We will present several studies and the previous literature related to our project which will help us in developing our own design and implementation in further stages. These studies illustrate some of the existing hardware and software solutions to design a parking system.

**A Secure Parking Reservation System Using GSM Technology**

[11] have used GSM technology to propose a secure and smart reservation system for parking areas. The system consists of two major modules which are security reservation module and parking lot monitoring.

Secure Reservation Module

The main idea is that the user has to send SMS message using GSM (Global System for Mobile communications**)** to a Visual Basic application in a laptop/PC. The VB then processes the requested data and checks the available spot. They used the available APIs in VB to process SMS messages. If the requested parking lot is available, the user will receive a confirmation message that has the location for the spot and password. Otherwise, it will receive a reject message.

Parking Lot Monitoring Module

The major aim of this module is to show the status of the parking spots using Parking Layout Animation program through Visual Basic application in a laptop/PC. Their program displays the status of each parking spot through colors. Green color for *empty,* yellow color for *reserved,* and red color for *occupied*. Initially the status of all parking spots are initialized as *empty*. The status will change to *reserved* if the user successfully reserves a parking spot. When the user arrives to the entrance gate, he enters the password then this password will be verified by the controller. If the password is valid, the gate will be opened, and the Parking Layout module will update the spot status to *occupied*.

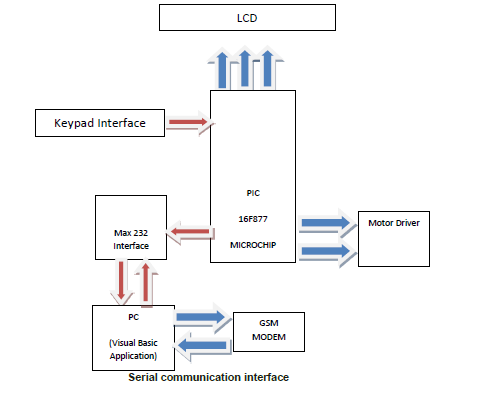
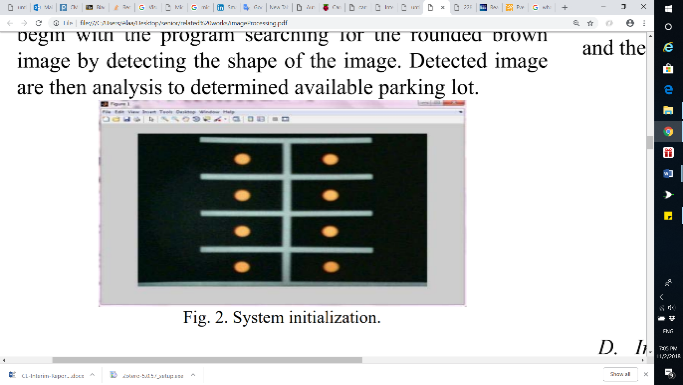
By using this approach, the user will be charged by telecommunication company in order to send SMS messages. Also, GSM is prone to be easily congested in crowded areas [12]. In their approach, all information is stored locally on one computer which is inefficient as the computer memory will eventually be full and not able to store any more information from the system. Also, if the computer is somehow damaged, all the data will be lost.

Figure ‎2‑1: A Secure Parking Reservation System Using GSM Technology – System architecture

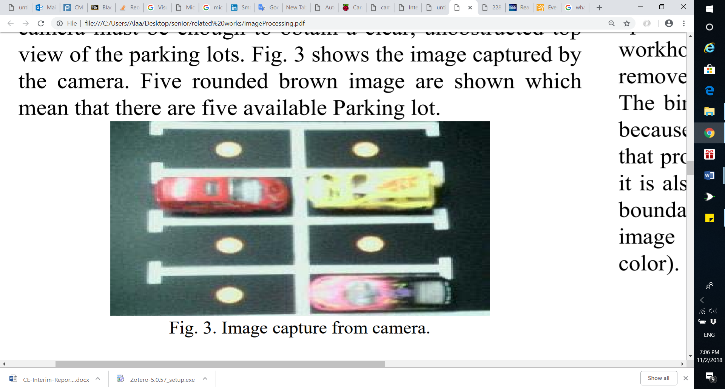
**Intelligent Parking Space Detection System Based on Image Processing**

[13] used image processing technique to detect if the parking lot is empty or not. The proposed solution is carried out in 5 steps as follows:

1- System Initialization:

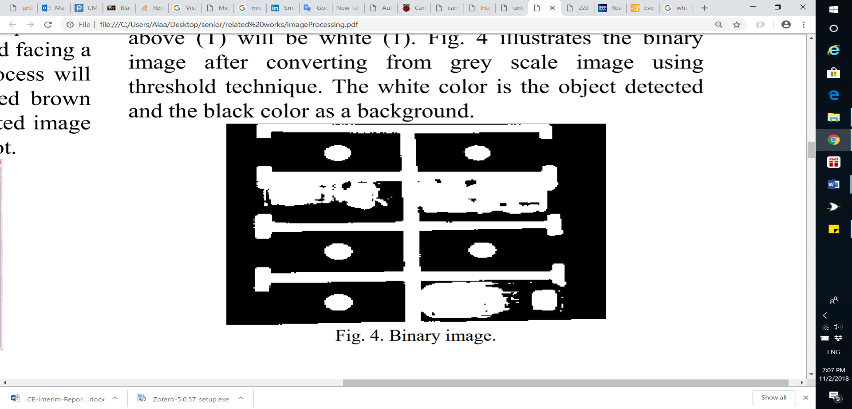
At the beginning a rounded brown image will be drawn at each parking manually. The benefit of this step is to determine the position of all parking spots in the image.

Figure ‎2‑2: System Initialization

2- Image Acquisition:

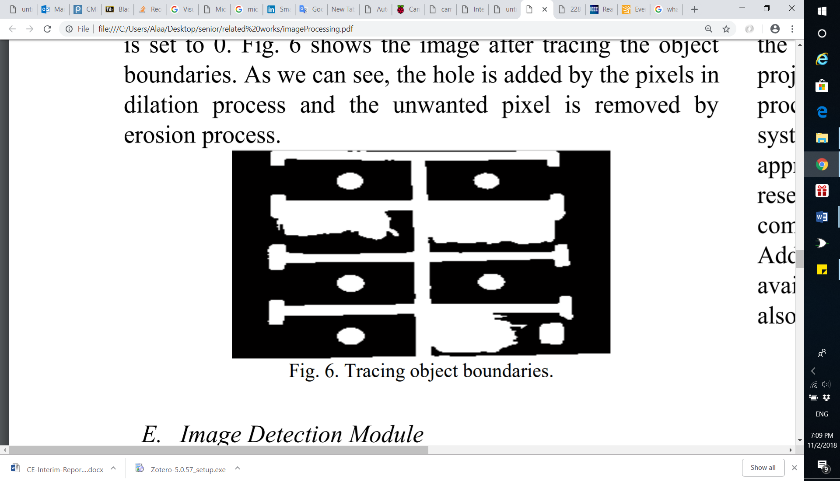
The camera will capture image for car park scene and send it to the processing unit that runs in MATLAB program. This step is known as image acquisition.

Figure ‎2‑3: Image capture from camera

3- Image Segmentation:

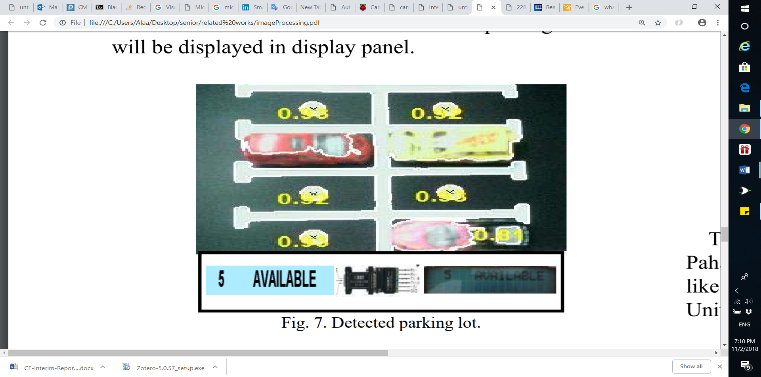
In the step RGB Image will be converted to grey scale image. Then applying thresholding technique on grey scale image to create the binary image. this technique is helping in separate the objects from the background.

Figure ‎2‑4: Binary image

4- Image Enhancement:

In this part the morphology functions such as dilation, erosion, opening and closing will be used to remove noise from the binary image. Also, these functions are used to trace the exterior boundaries of the object.

Figure ‎2‑5: Image after Tracing object boundaries

5- Image Detection Module:

In the last step the system will detect if the parking lot is empty or not by identify the rounded brown image at every spot.

Figure ‎2‑6: Detected parking spot

This approach may not function well in extreme weather conditions as the camera might be damaged. Also, camera must be placed in a suitable location where all parking areas are seen clearly and there are no objects that might obstructed the vision.

**Monitoring Parking Space Availability via Zigbee Technology**

In [14], the system’s main functionality is to allow its users to view the available parking spots through a screen at the gates of a parking area using Zigbee wireless technology for communication and digital infrared sensors for vehicle detection.

The system consists of two main modules:

Parking Lot Vacancy Monitoring Module

The module consists of digital infrared sensor, Zigbee module, PIC microcontroller 18F4550, and an LCD. This module is used to detect the available parking spots. To explain further, a digital infrared sensor is implemented on the top of each parking spot to detect the existence of a vehicle such that each sensor is used to monitor one parking spot. Infrared sensors work with reflected light wave and consist of an IR transmitter and an IR receiver. The IR transmitter emits infrared light, which gets reflected in case it met a reflecting surface (e.g. white color). The IR receiver then detects the reflected light and calculates the distance between the sensor and the object. The infrared sensor is interfaced with a PIC microcontroller, which is also interfaced with a Zigbee module for wireless transmission. When the sensor senses a vehicle in the parking spot, it informs the microcontroller, then the microcontroller informs the master module through the Zigbee module.

Master Module

The master module consists of GUI display shown through a PC or a laptop, and a Zigbee module. The GUI display, shown in Figure 2-7, allows the user to see exactly which parking spot is available. Whenever the Zigbee module receives data about a particular parking spot from the Parking Lot Vacancy Monitoring module, it informs the PC/laptop that it is interfaced with to update the status of a parking spot.

Furthermore, their system has several disadvantages. The infrared sensor is hugely dependent on light; as brighter surfaces are easier to detect than darker surfaces which the sensor does not detect. Hence, changing light conditions could give wrong outputs. Also, the system does not have a mobile for a remote user interface and is only restricted to a GUI interface at the gates of the parking lot.

The hardware architecture of the system is shown in Figure 2-8

Figure ‎2‑7: GUI Display on PC or laptop

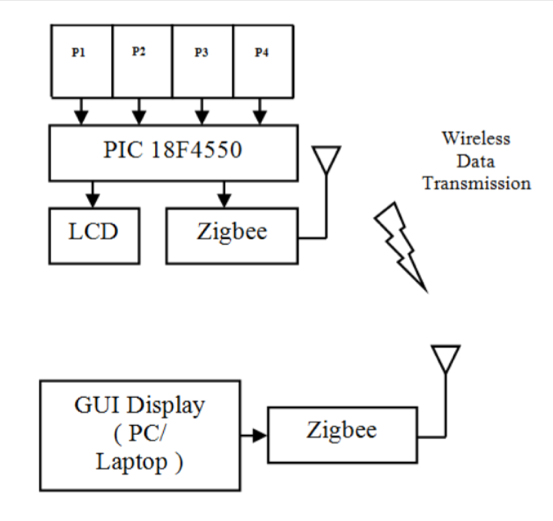


Figure ‎2‑8: Hardware architecture of the system

**Smart Parking System (SPS) Architecture Using Ultrasonic Detector**

[15] is a smart parking system for multilevel parking lots which uses ultrasonic sensors to detect the occupancy of a car and shows it to the user at the parking area using display boards. The display boards display the number of available parking spots, and they are implemented indoors -at each entrance to each level and at the end of each aisle of the parking lot- and outdoors -at the entrance and exit of the parking lot-. Furthermore, an ultrasonic sensor is applied on the ceiling of a parking spot and is used to detect cars on that spot only. LEDs with different colors accompany each ultrasonic sensor to inform the user of the parking spot status (reserved, occupied, vacant or handicapped). The system includes a monitoring software and line detection system to detect improper parking. The line detection system is done by adding two additional ultrasonic sensors horizontally on the right and left of each parking spot such that they face the parking spot lines. Whenever a car goes through the detection line, an alarm will go off until the car is moved out of the line.

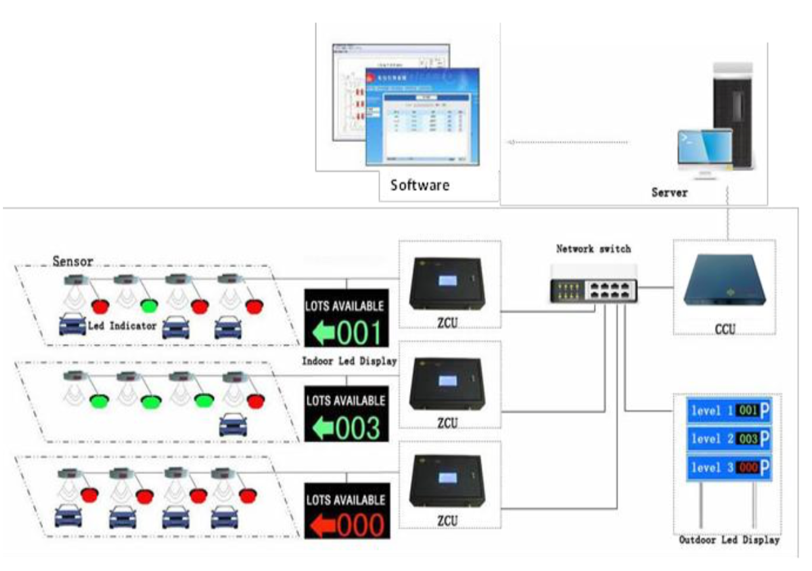


Figure ‎2‑9: Hardware architecture of the system

Figure 2-9 shows the overall hardware architecture of the system. The system consists of ultrasonic sensors, LED indicators, indoor display boards, outdoor display board(s), zone control unit (ZCU), central control unit (CCU), network switch, telephone cable and management software. When the ultrasonic sensor detects a change in the status of a parking area, it transmits the status to the zone control unit (ZCU) through a telephone cable. The ZCU then forwards the information to the central control unit (CCU) with Cat5 cables and sends commands to the indoor display boards and the LEDs on the parking sports to change status accordingly. Both the indoor display boards and the LEDs are connected to the ZCU through RS-485 ports. The ZCU is connected to the central control unit (CCU) through network switch and LAN connections. The CCU, upon receiving new information from any of the ZCUs, processes the data with the whole parking lot data, then transmits commands to the outdoor display boards to update the parking area information.

The system manages the multilevel parking area nicely. However, the system is wired with a lot of hardware components which has a huge number of cables. Any extension to the system would require more cables, and more complications on the system.

**Intelligent Parking System**

In [16], the system is based on image processing techniques which capture, and processes circles or rounded images drawn at parking lot and shown in an android application. This rounded image indicates the empty parking spaces to the users.

A camera is used with a sensor to take photos to show the occupancy of car parks. The information about whether a parking space is available or not is displayed on the application. If the space is available, a green circle is displayed to the user. As the camera detect the vacant space of parking and shown on the screen with green circles are placed on the vacant car parks. Green circle helps users so that users can easily differentiate whether a car is parked in a spot or not. So, if there is a car in a spot, no green circle will be shown at that spot.

The system contains three modules: Monitoring module, Control module and a displaying unit

The monitoring module has ultrasonic sensors which identifies the free parking spaces and transmits the Information to control unit which process/send information to the administrative system. Besides that, there’s a centralized system for supervisory receives information of parking space from the controller. It then sends the information to the user’s phone.

Below (Figure 2-10) illustrate the working process of the system

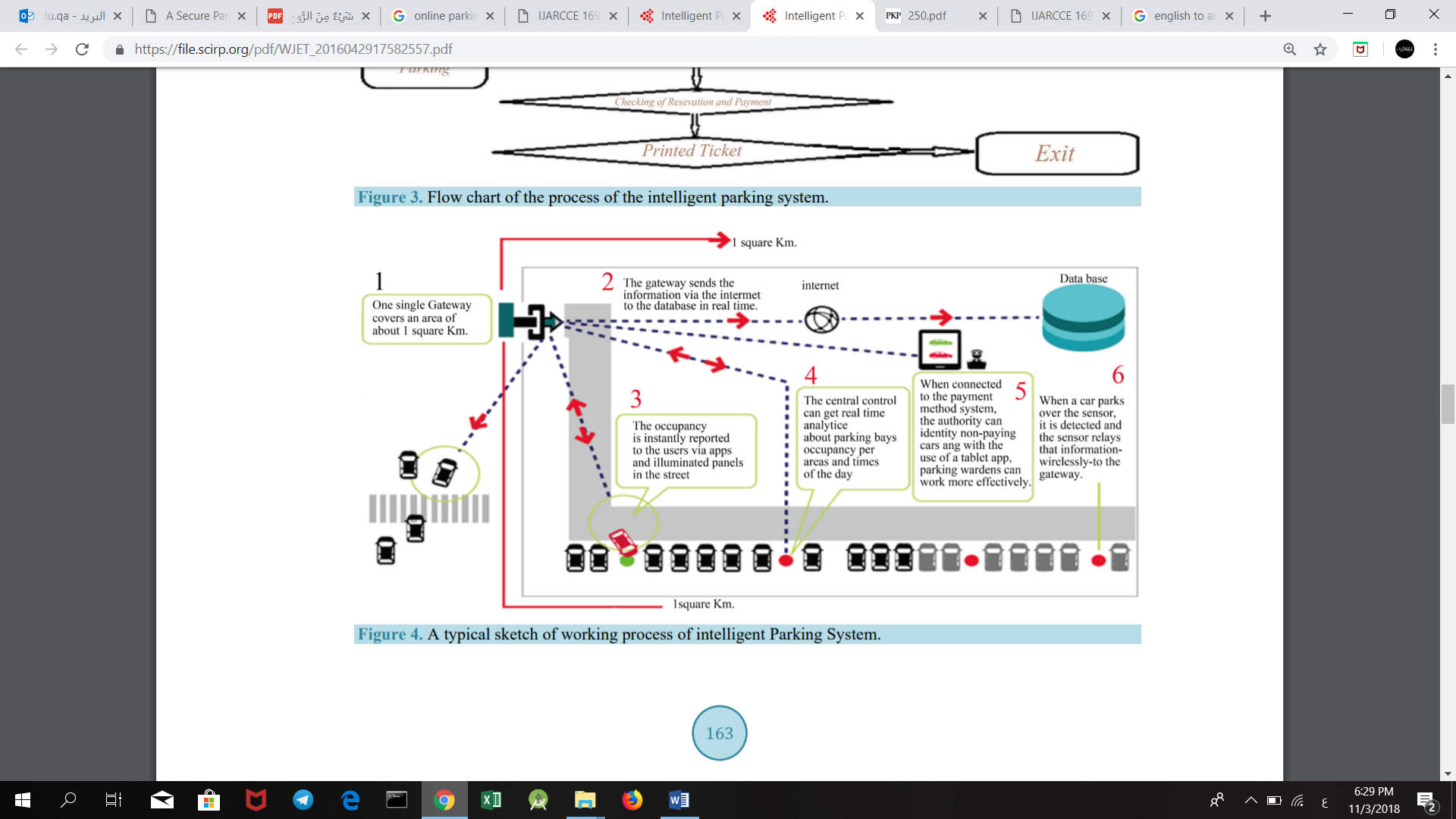


Figure ‎2‑10: Working process of the system

Related work comparison

Table ‎2‑1: Related work comparison

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Paper** | **Hardware Used** | | | **Service** | | | **User Interface Through** | **Connection** |
| **Microcontroller** | **Sensors** | **Others** | **Reservation** | **Availability checking** | **Payment** |
| **[11]** | Pic Microcontroller |  | LCD,  Motor driver |  |  |  | Visual Basic Application | GSM |
| **[13]** |  |  | Camera |  |  |  |  |  |
| **[14]** | Pic Microcontroller | Infrared Sensor | Zigbee- module  LCD |  |  |  | GUI displayed on PC or laptop | Wireless |
| **[15]** |  | Ultrasonic Sensor | Display  Boards,  LEDs,  Network Switch |  |  |  |  | Telephone cables, LAN connection |
| **[16]** |  | Ultrasonic sensors | Camera |  |  |  | Android Mobile Application |  |
| **Our Method** | Arduino Uno | Ultrasonic  RFID | Servo  motor,  LEDs |  |  |  | Mobile- Application  Website | Wireless |

Having compared the existing solutions to ours in Table XX, the following remarks come to mind. All the related works implement one of the following services (Checking Availability, Reservations, Payment) except [16]. [16] provides all the above services as we aim to do in our system. However, [16] only provides an Android application while we will provide in addition to that a website. Furthermore, the above approaches use ultrasonic, infrared and camera for car detection only. Our approach uses ultrasonic sensor for car detection and RFID (Radio Frequency Identification) for car identification and detection.

# Requirements analysis

In this section, the software development process used in this project is defined. The functional and nonfunctional requirements of the system are addressed.

## Software development process

The software development process in our project is Scrum. Scrum is an iterative and incremental agile software development process that focuses on adapting to rapid changes. Scrum was chosen for our project because it does not require us to know exactly how and what to do from the beginning of the development process. The Scrum process is responsive to changes and improvements with repeated reviews and work inspection [17]. Figure 3-1 shows the Scrum process overview

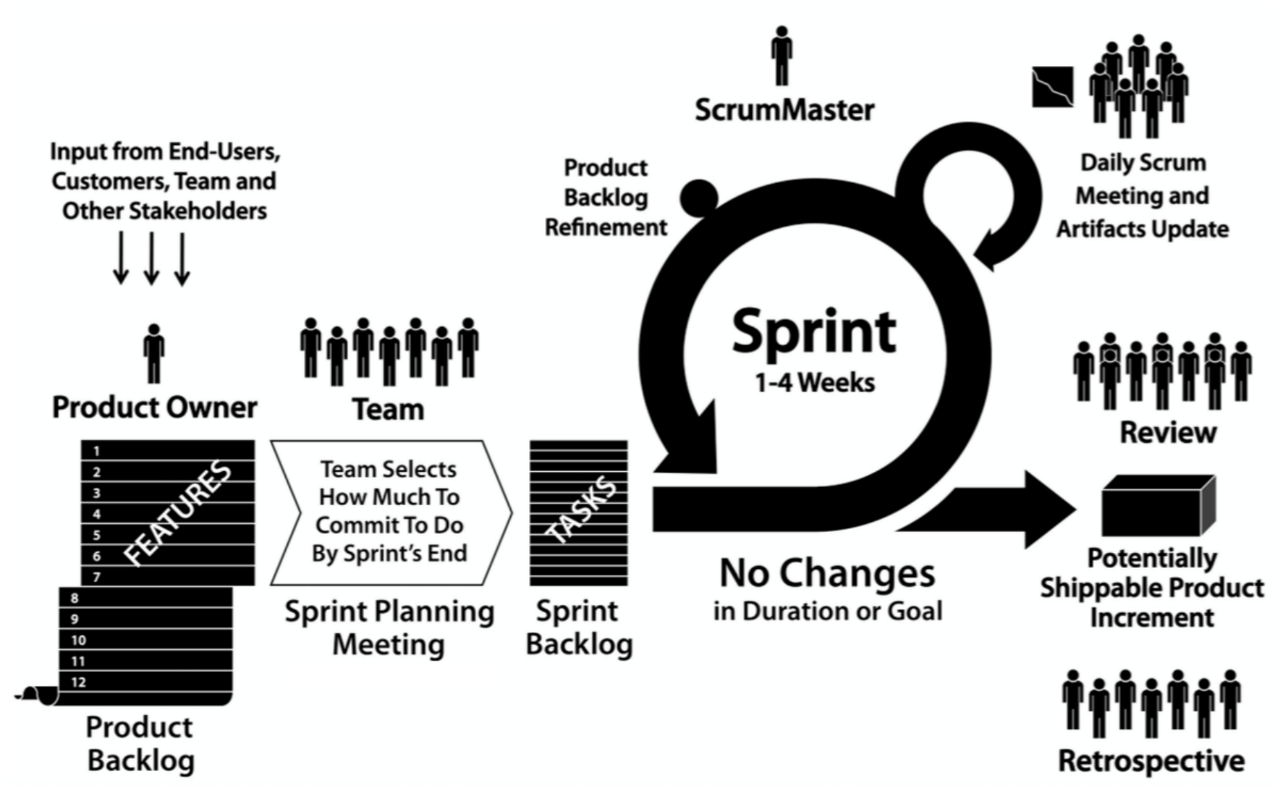


Figure ‎3‑1: Scrum process overview

The main part of the Scrum cycle is the Sprint. Each Sprint is time restricted to one week and has 4 steps as shown in Figure 3-2.

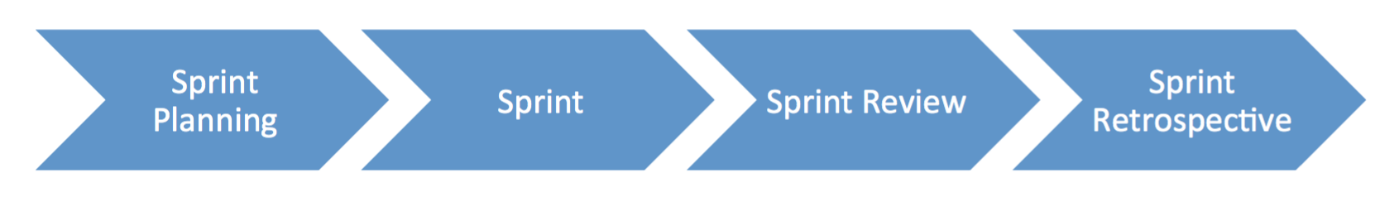


Figure ‎3‑2: Scrum cycle

**1. Sprint Planning**

At the beginning of the project, we collected the requirements into a Product Backlog from the users by researching the parking problem in Qatar University and the users’ needs. Before starting a Sprint, we hold up a meeting to decide on the top priority requirements from the Product Backlog and create a Sprint Backlog. The Sprint Backlog contains tasks that we think will be done during the Sprint which is in our case one week.

**2. Sprint**

In this stage, we start working on the tasks as specified in the Sprint Backlog. In addition, a Daily Scrum Meeting is held to discuss the tasks that was done by each member so far, and what will be done before the next meeting. Also, if any member has faced any difficulties during the task, we try to find a solution for it.

**3. Sprint Review**

At the end of a Sprint, a meeting is held with the ScrumMaster (supervisor) to demonstrate what was done. The ScrumMaster suggest solutions to the issues we faced during the Sprint.

**4. Sprint Retrospective**

After the Sprint Review, we refine our tasks to meet the comments and suggestions given by the ScrumMaster and decide on the improvements to be done to make the next Sprint smoother.

## Functional requirements

* An RFID reader should be installed and kept operational at the gate of the parking facility.
* VIP users must have an RFID tag in their possession before entering the parking facility.
* Sensors should be hardwired to Arduino.
* Arduino must be connected to a power source.
* Arduino must be connected to a Wi-Fi module.
* Firebase must have a connection with the Arduinos, the mobile application and the website via Wi-Fi.
* The mobile must have internet connection.
* The user should be a university member or visitor.

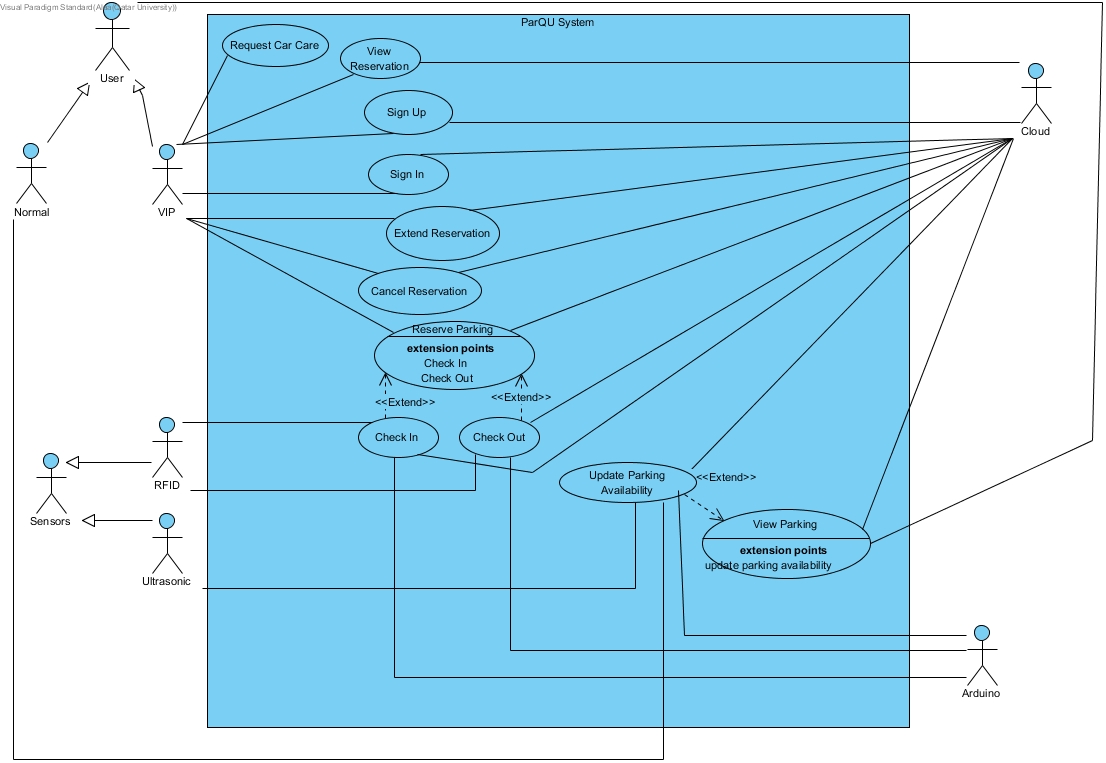


Figure ‎3‑3: Use case diagram

Table ‎3‑1: Actor type and descriptions

|  |  |
| --- | --- |
| Actor types | Description |
| VIP User | Anyone who subscribed to the VIP services such as reservation service and request car care service. |
| Normal User | Anyone who is not subscribed to the VIP services. |

Table ‎3‑2: Use cases and descriptions

|  |  |
| --- | --- |
| Use case | Brief description |
| Sign Up | VIP user registers in the system. |
| Sign In | VIP user logs into the system. |
| Reserve Parking | VIP user reserves a parking spot for several hours. |
| View Reservation | VIP user views reservation details. |
| Extend Reservation | The VIP user extends a reservation. |
| Cancel Reservation | The VIP user cancels a reservation. |
| Check In | The system reads the RFID tag on the car, checks its validity, then allows reserved VIP users only to enter the parking lot. |
| Check Out | The system reads the RFID tag on the car, checks its validity, then update available parking spots. |
| View Parking | The user views a map with the current status of parking spots. |
| Update Parking Availability | The system receives updated data from sensors through the Arduino board. |
| Request Car Care | The VIP user request car care service which will display contact numbers of car care company. |

## Non-functional requirements

**Design constraints**

Table ‎3‑3: Technical design constraints

|  |  |
| --- | --- |
| Name | Description |
| Reliability | Data is saved at all times on the Firebase database, so no data will be lost. |
| Scalability | The system can support the addition of sensors and components as well as having more users and the cloud can be upgraded accordingly. |
| Connectivity | Firebase database needs to be connected to the internet to collect updated data from the sensors through the Arduino board. Also, the mobile application needs to be connected to the internet to get the data |
| Availability | The system should be always available. During using the application, if any data changed in firebase database, the application should automatically display the updates without needing to refresh the page. |
| Mobility | The system can be accessed from two different platforms: an Android application and a website |
| Power | Power source needed for:  Motor: 3-7V  Sensor: 5V  Arduino: 7-12V |

Table ‎3‑4: Practical design constraints

|  |  |  |
| --- | --- | --- |
| Type | Name | Description |
| Economics | Cost | The prototype for availability module which consists of 4 parking spots should cost on average 270 QR.  The prototype for reservation module should cost on average 260 QR. |
| Social | Usability | A normal user with minimal software knowledge should be able to use the mobile application and the website with ease |
| Sustainability | Maintenance | The system’s component should be easy to replace, remove and implement. |
| Quality | Performance | The system should provide efficient information and accurate readings from the parking area. |

**Design standards**

Table ‎3‑5: Standard types and descriptions

|  |  |
| --- | --- |
| Standard Types | Description |
| Networking Standard | HTTP (Hypertext transfer protocol) |
| Security Standard | OpenID Connect (used for delegated authentication) |
| Wireless Standard | IEEE 802.11 (Wi-Fi) |
| Serial Communication Standards | SPI (Serial Peripheral Interface) |

## Assumptions

Below are some assumptions that are out of the scope of the work done in this project but necessary for the project to work properly:

* Arduino is always connected to the sensors.
* A power source is connected to the Arduino.
* Arduino always has a Wi-Fi connection.
* Firebase can always store data without limits.
* Application and website are always available.
* Mobile and PC that access our application and website is always connected to the internet.
* VIP user always has an RFID tag.

## Ethics

Table ‎3‑6: IEEE and ACM code of ethics

|  |  |
| --- | --- |
| Code | Project Perspective |
| IEEE  To seek, accept, and offer honest criticism of technical work, to acknowledge and correct errors, and to credit properly the contributions of others.  ACM  - Honor property rights including copyrights and patent.  - Give proper credit for intellectual property. | Any project or work used is referenced in this paper. Feedback will be accepted by the supervisor and the examiners. |
| IEEE  To treat fairly all persons and to engage in acts of discrimination based on race, religion, gender, disability, age, national origin, sexual orientation, gender identify, or gender expression.  ACM  - Be fair and take action not to discriminate. | Project members have all worked together to produce this work which can be used by all types of people. |
| IEEE  To assist colleagues and co-workers in their professional development and to support them in following this code of ethics. | The project is collaborative; every team member will work on it and ensure the code of ethics is strictly followed. |
| ACM  - Contribute to society and human well-being.  - Manage personnel and resources to design and build information systems that enhance the quality of working life. | The design aims to achieve better standards and benefits. |