AUTOMATED SURVEILLANCE SYSTEM FOR VEHICLE VERIFICATION USING NUMBER PLATE RECOGNITION IN WILDFOREST

A PROJECT REPORT

Submitted by

K. ASWIN KUMAR	**912419104003**
R. KARTHIK	**912419104013**

in partial fulfillment for the award of the degree of

BACHELOR OF ENGINEERING

IN

COMPUTER SCIENCE AND ENGINEERING

SHANMUGANATHAN ENGINEERING COLLEGE

ARASAMPATTI

ANNA UNIVERSITY :: CHENNAI 600 025

MAY 2023

BONAFIDE CERTIFICATE

Certified that this project report " **AUTOMATED SURVEILLANCE SYSTEM FOR VEHICLE VERIFICATION USING NUMBER PLATE RECOGNITION IN WILDFORESTC** " is the bonafide work of **"K. ASWIN KUMAR (912419104003), R. KARTHIK (912419104013)** " who carried out the project under my supervision.

HEAD OF THE DEPARTMENT	**SUPERVISOR**
Mrs. R. KAVITHA, M.Tech.,	**Mrs. R. KAVITHA, M.Tech.,**
Assistant Professor,	Assistant Professor,
Department of Computer Science	Department of Computer Science
and Engineering,	and Engineering,
Shanmuganathan Engineering	Shanmuganathan Engineering
College,	College,
Arasampatti-622507.	Arasampatti-622507.

Submitted for the viva voce to be held on _______________

INTERNAL EXAMINER **EXTERNAL EXAMINER**

ACKNOWLEDGEMENT

"HARDWORK NEVER FAILS" So we thank god for having gracefully blessed us to come up till now and thereby giving strength and courage to complete the project successfully. We sincerely submit this project to the almighty lotus feet.

We wish to acknowledge with thanks to the significant contribution given by the management of our college chairman **"KALLVI VALLAL Mrs. Pichappa Valliammai"**, Correspondent **Dr. P. Manikandan**, and Secretary **Mr. M. Vishwanathan**, Shanmuganathan Engineering College, Arasampatti, for their extensive support.

We convey our indebted humble thanks to our energetic Principal **Dr. KL. Muthuramu M.E., (W.R), M.E.(S.E), M.I.S.T.E., F.I.E., Ph.D.,** for his moral support to complete this project.

We are grateful to our Head of the Department **Mrs. R. Kavitha M.Tech.,** for her valuable guidelines during the courses of the project.

Gratitude will never fail towards our dynamic and effective internal guide **Mrs. R. KAVITHA, M.Tech.,** for her unending help that have been provided to us throughout the project

We wish to acknowledge the department project coordinator **Mr. V. Ilayaraja, M.E.,** for his valuable, innovative suggestion, constructive instruction, constant encouragement, and during project proceedings

We also convey our humble thanks to all the staff members of the CSE Department who had provided their contribution to steer our project towards the successful completion.

Finally, we thank our family members who have spent their wealth for our studies and have motivated us to complete this project successfully.

ABSTRACT

Forest entry access refers to the process of controlling and regulating the entry of vehicles into a forest area. This is typically done to manage the impact of human activity on the forest environment, prevent illegal activities such as poaching, logging, or dumping, and ensure the safety of visitors. There are various methods used to control forest entry access, including physical barriers such as gates, guards, and fences, as well as technology-based solutions such as RFID and NPR. Physical barriers can be effective, but they require human presence to operate, which can be costly and resource-intensive. Technology-based solutions, on the other hand, offer a more efficient and automated way to control access. Number plate recognition (NPR) technology has been increasingly used for various purposes, including access control. When combined with RFID technology, NPR can be used to control access to forests. The system involves installing a camera at the entrance of the forest that captures the number plate of every vehicle entering and leaving the forest. The camera sends the image of the number plate to an OCR software that recognizes the characters on the plate. The OCR software then compares the license plate number with a database of registered vehicles allowed to access the forest. The database can be updated manually or automatically using RFID tags. If the license plate number matches a registered vehicle, the gate opens, and the vehicle is allowed to enter the forest.

KEYWORDS: Forest entry system, Number plate detection, Number plate Recognition, Character segmentation, Optical Character recognition.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF ABBREVIATIONS

1. RNN - RECURRENT NEURAL NETWORKS
2. CNN - CONVOLUTIONAL NEURAL NETWORKS
3. NPR - NUMBER PLATE RECOGNITION
4. OCR - OPTICAL CHARACTER RECOGNITION
5. NLP - NATURAL LANGUAGE PROCESSING
6. ALPR - AUTOMATIC LICENSE PLATE RECOGNITION

CHAPTER 1

INTRODUCTION

1.1 ARTIFICIAL INTELLIGENCE

AI (artificial intelligence) is the simulation of human intelligence processes by machines, especially computer systems. These processes include learning (the acquisition of information and rules for using the information), reasoning (using rules to reach approximate or definite conclusions) and self-correction. Particular applications of AI include expert systems, speech recognition and machine vision.

AI can be categorized in any number of ways, but here are two examples. The first classifies AI systems as either weak AI or strong AI. Weak AI, also known as narrow AI, is an AI system that is designed and trained for a particular task. Virtual personal assistants, such as Apple's Siri, are a form of weak AI. Strong AI, also known as artificial general intelligence, is an AI system with generalized human cognitive abilities so that when presented with an unfamiliar task, it has enough intelligence to find a solution.

The Turing Test, developed by mathematician Alan Turing in 1950, is a method used to determine if a computer can actually think like a human, although the method is controversial.

1.2 TECHNOLOGIES IN AI

1.2.1 Automation

What makes a system or process function automatically? For example, robotic process automation (RPA) can be programmed to perform high-volume, repeatable tasks that humans normally perform. RPA is different from IT automation in that it can adapt to changing circumstances.

1.2.2 Machine learning

The science of getting a computer to act without programming. Deep learning is a subset of machine learning that, in very simple terms, can be thought of as the automation of predictive analytics. There are three types of machine learning algorithms:

1. Supervised learning: Data sets are labeled so that patterns can be detected and used to label new data sets
2. Unsupervised learning: Data sets aren't labeled and are sorted according to similarities or differences
3. Reinforcement learning: Data sets aren't labeled but, after performing an action or several actions, the AI system is given feedback

1.2.3 Machine vision

The science of allowing computers to see. This technology captures and analyzes visual information using a camera, analog-to-digital conversion and digital signal processing. It is often compared to human eyesight, but machine vision isn't bound by biology and can be programmed to see through walls, for example.

It is used in a range of applications from signature identification to medical image analysis. Computer vision, which is focused on machine-based image processing, is often conflated with machine vision.

1.2.4 Natural language processing (NLP)

The processing of human -- and not computer -- language by a computer program. One of the older and best-known examples of NLP is spam detection, which looks at the subject line and the text of an email and decides if it's junk. Current approaches to NLP are based on machine learning. NLP tasks include text translation, sentiment analysis and speech recognition.

1.2.5 Robotics

A field of engineering focused on the design and manufacturing of robots. Robots are often used to perform tasks that are difficult for humans to perform or perform consistently. They are used in assembly lines for car production or by NASA to move large objects in space. Researchers are also using machine learning to build robots that can interact in social settings.

1.2.6 Self-driving cars

These use a combination of computer vision, image recognition and deep learning to build automated skill at piloting a vehicle while staying in a given lane and avoiding unexpected obstructions, such as pedestrians**.**

1.3 APPLICATIONS OF AI

Artificial intelligence has made its way into a number of areas. Here are six examples.

1. **AI in Healthcare**:

The biggest bets are on improving patient outcomes and reducing costs. Companies are applying machine learning to make better and faster diagnoses than humans. One of the best-known healthcare technologies is IBM Watson. It understands natural language and is capable of responding to questions asked of it.

The system mines patient data and other available data sources to form a hypothesis, which it then presents with a confidence scoring schema. Other AI applications include chatbots, a computer program used online to answer questions and assist customers, to help schedule follow-up appointments or aid patients through the billing process, and virtual health assistants that provide basic medical feedback.

2. **AI in Business**:

Robotic process automation is being applied to highly repetitive tasks normally performed by humans. Machine learning algorithms are being integrated into analytics and CRM platforms to uncover information on how to better serve customers. Chatbots have been incorporated into websites to provide immediate service to customers. Automation of job positions has also become a talking point among academics and IT analysts.

3. **AI in Education**:

AI can automate grading, giving educators more time. AI can assess students and adapt to their needs, helping them work at their own pace. AI tutors can provide additional support to students, ensuring they stay on track. AI could change where and how students learn, perhaps even replacing some teachers.

4. **AI in Finance**:

AI in personal finance applications, such as Mint or TurboTax, is disrupting financial institutions. Applications such as these collect personal data and provide financial advice. Other programs, such as IBM Watson, have been applied to the process of buying a home. Today, software performs much of the trading on Wall Street.

5. **AI in Law**:

The discovery process, sifting through documents, in law is often overwhelming for humans. Automating this process is a more efficient use of time. Startups are also building question-and-answer computer assistants that can sift programmed-to-answer questions by examining the taxonomy and ontology associated with a database.

6. **AI in Manufacturing**:

This is an area that has been at the forefront of incorporating robots into the workflow. Industrial robots used to perform single tasks and were separated from human workers, but as the technology advanced that changed.

1.4 DISADVANTAGES OF AI

While AI tools present a range of new functionality for businesses, artificial intelligence also raises some ethical questions. Deep learning algorithms, which underpin many of the most advanced AI tools, only know what's in the data used during training. Most available data sets for training likely contain traces of human bias. This in turn can make the AI tools biased in their function.

This has been seen in the Microsoft chatbot Tay, which learned a misogynistic and anti-Semitic vocabulary from Twitter users, and the Google Photo image classification tool that classified a group of African Americans as gorillas. The application of AI in the realm of self-driving cars also raises ethical concerns. When an autonomous vehicle is involved in an accident, liability is unclear.

Autonomous vehicles may also be put in a position where an accident is unavoidable, forcing it to make ethical decisions about how to minimize damage. Another major concern is the potential for abuse of AI tools. Hackers are starting to use sophisticated machine learning tools to gain access to sensitive systems, complicating the issue of security beyond its current state. Deep learning-based video and audio generation tools also present bad actors with the tools necessary to create so-called deep fakes, convincingly fabricated videos of public figures saying or doing things that never took place.

1.5 IMAGE PROCESSING:

In imaging science, image processing is processing of images using mathematical operations by using any form of signal processing for which the input is an image, a series of images, or a video, such as a photograph or video frame; the output of image processing may be either an image or a set of characteristics or parameters related to the image.

Most image-processing techniques involve treating the image as a two-dimensional signal and applying standard signal-processing techniques to it. Images are also processed as three-dimensional signals with the third-dimension being time or the z-axis. Image processing usually refers to digital image processing, but optical and analog image processing also are possible.

This project is about general techniques that apply to all of them. The acquisition of images (producing the input image in the first place) is referred to as imaging. Closely related to image processing are computer graphics and computer vision.

1.6 DEEP LEARNING

Deep learning is an artificial intelligence (AI) function that imitates the workings of the human brain in processing data and creating patterns for use in decision making. Deep learning is a subset of machine learning in artificial intelligence that has networks capable of learning unsupervised from data that is unstructured or unlabeled. Also known as deep neural learning or deep neural network.

- Deep learning is an AI function that mimics the workings of the human brain in processing data for use in detecting objects, recognizing speech, translating languages, and making decisions.
- Deep learning AI is able to learn without human supervision, drawing from data that is both unstructured and unlabeled.
- Deep learning, a form of machine learning, can be used to help detect fraud or money laundering, among other functions.

Deep learning has evolved hand-in-hand with the digital era, which has brought about an explosion of data in all forms and from every region of the world. This data, known simply as big data, is drawn from sources like social media, internet search engines, e-commerce platforms, and online cinemas, among others. This enormous amount of data is readily accessible and can be shared through fintech applications like cloud computing.

However, the data, which normally is unstructured, is so vast that it could take decades for humans to comprehend it and extract relevant information. Companies realize the incredible potential that can result from unraveling this wealth of information and are increasingly adapting to AI systems for automated support. One of the most common AI techniques used for processing big data is machine learning, a self-adaptive algorithm that gets increasingly better analysis and patterns with experience or with newly added data.

If a digital payments company wanted to detect the occurrence or potential for fraud in its system, it could employ machine learning tools for this purpose. The computational algorithm built into a computer model will process all transactions happening on the digital platform, find patterns in the data set, and point out any anomaly detected by the pattern. Deep learning, a subset of machine learning, utilizes a hierarchical level of artificial neural networks to carry out the process of machine learning. The artificial neural networks are built like the human brain, with neuron nodes connected together like a web.

While traditional programs build analysis with data in a linear way, the hierarchical function of deep learning systems enables machines to process data with a nonlinear approach. A traditional approach to detecting fraud or money laundering might rely on the amount of transaction that ensues, while a deep learning nonlinear technique would include time, geographic location, IP address, type of retailer, and any other feature that is likely to point to fraudulent activity.

The first layer of the neural network processes a raw data input like the amount of the transaction and passes it on to the next layer as output. The second layer processes the previous layer's information by including additional information like the user's IP address and passes on its result. The next layer takes the second layer's information and includes raw data like geographic location and makes the machine's pattern even better. This continues across all levels of the neuron network. Artificial neural networks (ANNs) were inspired by information processing and distributed communication nodes in biological systems. ANNs have various differences from biological brains. Specifically, neural networks tend to be static and symbolic, while the biological brain of most living organisms is dynamic (plastic) and analogue. The adjective "deep" in deep learning refers to the use of multiple layers in the network. Early work showed that a linear perceptron cannot be a universal classifier, and then that a network with a nonpolynomial activation function with one hidden layer of unbounded width can on the other hand so be.

Deep learning is a modern variation which is concerned with an unbounded number of layers of bounded size, which permits practical application and optimized implementation, while retaining theoretical universality under mild conditions. In deep learning the layers are also permitted to be heterogeneous and to deviate widely from biologically informed connectionist models, for the sake of efficiency, trainability and understandability, whence the "structured" part.

Deep Learning is a machine learning technique that constructs artificial neural networks to mimic the structure and function of the human brain. In practice, deep learning, also known as deep structured learning or hierarchical learning, uses a large number hidden layers -typically more than 6 but often much higher - of nonlinear processing to extract features from data and transform the data into different levels of abstraction (representations).

As an example, assume the input data is a matrix of pixels. The first layer typically abstracts the pixels and recognizes the edges of features in the image. The next layer might build simple features from the edges such as leaves and branches. The next layer could then recognize a tree and so on. The data passing from one layer to the next is considered a transformation, turning the output of one layer into the input for the next.

Each layer corresponds with a different level of abstraction and the machine can learn which features of the data to place in which layer/level on its own. Deep learning is differentiated from traditional “shallow learning” because it learns much deeper levels of hierarchical abstraction and representations. This learning technique is a groundbreaking tool for processing large quantities of data, since the performance of the machine improves as it analyzes more data. As the amount of data increases, the machine becomes more adept at recognizing even hidden patterns among the data. Because the machine is also learning from the processed data, it is able to perform feature extraction and abstraction automatically from the raw data with little to no human input.

1.7 DEEP LEARNING IN IMAGE PROCESSING

In imaging science, image processing is processing of images using mathematical operations by using any form of signal processing for which the input is an image, a series of images, or a video, such as a photograph or video frame; the output of image processing may be either an image or a set of characteristics or parameters related to the image. Most image-processing techniques involve treating the image as a two-dimensional signal and applying standard signal-processing techniques to it. Images are also processed as three-dimensional signals with the third-dimension being time or the z-axis. Image processing usually refers to digital image processing, but optical and analog image processing also are possible.

This project is about general techniques that apply to all of them. The acquisition of images (producing the input image in the first place) is referred to as imaging. Closely related to image processing are computer graphics and computer vision. In computer graphics, images are manually made from physical models of objects, environments, and lighting, instead of being acquired (via imaging devices such as cameras) from natural scenes, as in most animated movies. Computer vision, on the other hand, is often considered high-level image processing out of which a machine/computer/software intends to decipher the physical contents of an image or a sequence of images (e.g., videos or 3D full-body magnetic resonance scans).

In modern sciences and technologies, images also gain much broader scopes due to the ever-growing importance of scientific visualization (of often large-scale complex scientific/experimental data). Examples include microarray data in genetic research, or real-time multi-asset portfolio trading in finance. Image analysis is the extraction of meaningful information from images; mainly from digital images by means of digital image processing techniques. Image analysis tasks can be as simple as reading bar coded tags or as sophisticated as identifying a person from their face.

Computers are indispensable for the analysis of large amounts of data, for tasks that require complex computation, or for the extraction of quantitative information. On the other hand, the human visual cortex is an excellent image analysis apparatus, especially for extracting higher-level information, and for many applications - including medicine, security, and remote sensing — human analysts still cannot be replaced by computers. For this reason, many important image analysis tools such as edge detectors and neural networks are inspired by human visual perception models. Image editing encompasses the processes of altering images, whether they are digital photographs, traditional photochemical photographs, or illustrations.

Traditional analog image editing is known as photo retouching, using tools such as an airbrush to modify photographs, or editing illustrations with any traditional art medium. Graphic software programs, which can be broadly grouped into vector graphics editors, raster graphics editors, and 3D modelers, are the primary tools with which a user may manipulate, enhance, and transform images. Many image editing programs are also used to render or create computer art from scratch. Raster images are stored in a computer in the form of a grid of picture elements, or pixels. These pixels contain the image's color and brightness information.

Image editors can change the pixels to enhance the image in many ways. The pixels can be changed as a group, or individually, by the sophisticated algorithms within the image editors. This article mostly refers to bitmap graphics editors, which are often used to alter photographs and other raster graphics. However, vector graphics software, such as Adobe Illustrator, CorelDRAW, Xara Designer Pro, PixelStyle Photo Editor, Inkscape or Vector, are used to create and modify vector images, which are stored as descriptions of lines, Bézier curves, and text instead of pixels. It is easier to rasterize a vector image than to vectorize a raster image; how to go about vectorizing a raster image is the focus of much research in the field of computer vision. Vector images can be modified more easily, because they contain descriptions of the shapes for easy rearrangement. They are also scalable, being factorizable at any resolution.

1.8 ADVANTAGES FOR AI

Artificial intelligence (AI) offers numerous advantages across a wide range of industries and applications. Some of the key advantages of AI include:

Increased efficiency: AI can automate tedious and repetitive tasks, allowing humans to focus on more complex and creative work. This can lead to significant increases in productivity and efficiency.

Improved Accuracy: AI systems can process and analyze vast amounts of data, enabling them to make highly accurate predictions and decisions. This can be particularly useful in fields such as healthcare, where AI can help diagnose diseases and develop personalized treatment plans.

Enhanced Personalization: AI can be used to analyze large datasets to identify patterns and preferences, enabling personalized experiences in areas such as e-commerce, entertainment, and social media.

Cost Savings: AI can help reduce costs by automating tasks that would otherwise require human labor, such as customer service or data entry.

Improved Safety: AI can be used to monitor and detect potential safety issues in real-time, such as in autonomous vehicles or industrial settings.

Increased Accessibility: AI can enable greater accessibility for individuals with disabilities by providing assistive technologies such as speech recognition or computer vision.

CHAPTER 2

2.LITERATURE SURVEY

2.1 TITLE: Random-Positioned License Plate Using Hybrid Broad Learning System and Convolutional Networks

AUTHOR: C.L. Philip Chen

In this paper, we propose a new method for license plate recognition. The proposed fully convolutional network is aimed for random-positioned object detection, and it is implemented by a fusion of multi-scale and hierarchical feature maps. We applied this network to license plate detection and achieved remarkable results. It not only detects random-positioned license plates but also predicts the location of desired objects when two similar license plates are attached on a vehicle together. This can be extended to detect other rigid objects as long as the train data is prepared. Second, this paper develops a simple and effective strategy for character segmentation, which detects key character and uses it as a reference for slant estimation and vertical projection. To distinguish similar characters such as letter 'B' and digit '8', letter 'O' and digit '0', we explore a dual-SAE_BLS structure for the recognition of letters and digits, respectively. All these techniques contribute to the promising results in accuracy and comparable processing speed. It is demonstrated by the comparisons with Faster-RCNN, PlateRecognizer, and other approaches at all stages of ALPR. In the future, it would be interesting to integrate these modules and apply our method to other regions that have similar situations as Macau. Each license plate recognition system initially aims to solve the local ALPR problem. The proposed method and other approaches are still subject to the certain limitations of regions. The license plate detection part and character recognition part can be applied to a new country or region with little change, but the segmentation part needs to be designed according to the standard of regional license plates. Best parameter settings are always problem-dependent.

2.1.1 ADVANTAGES

- Image based character recognition.

2.1.2 DISADVANTAGES

- Detects key character and uses it as a reference for slant estimation and vertical projection.

2.2 TITLE: Automated License Plate Recognition: A Survey On Methods and Techniques

AUTHOR: Jithmi Shashirangan

Automatic License Plate Recognition (ALPR) systems are attracting an increasing interest due to their applicability in intelligent transportation systems that have been installed in many countries for tasks such as traffic law enforcement and traffic monitoring. Besides, ALPR systems are also used to manage exit and entrance in vehicle parks, collect toll payments, and to control security measures in restricted areas like military campsites, and protected sanctuaries. Often, these ALPR systems are employed to prevent fraud and to intensify security in specific areas. For instance, they can be helpful when searching for missing vehicles or vehicles related to crimes. Unless for ALPR systems, this task requires a sizable amount of labour, time, and resources. Also, manual intervention in such tasks may lead to erroneous interpretations, and in the meantime, it is practically difficult for a human to remember or to read a license plate of a moving vehicle efficiently. Most of the ALPR systems are designed to deploy outdoors. However, it is challenging to detect and recognize license plates under changing environmental and weather conditions. These factors include illumination changes, snow or fog weather conditions, and day and night. Besides, there may exist issues related to the cameras and license plate variations. For instance, dust and vibrations of the camera may result in a blurry image, which makes the recognition task problematic and produce erroneous output.

2.2.1 ADVANTAGES

- The single-stage deep learning-based solutions have shown high performances with diverse datasets.

2.2.2 DISADVANTAGES

- Need a large number of benchmark datasets.

2.3 TITLE: Research On License Plate Recognition Algorithms Based On Deep Learning in Complex Environment

AUTHOR: Wang Weihong

The license plate recognition system has two main tasks, one is to locate the license plate, and the other is to identify the license plate characters. Recently, some people have proposed a two-stage algorithm combining with sequence recognition, therefore the recognition process can only be divided into license plate location and character recognition, leaving out character segmentation. Character segmentation is often used in traditional text recognition algorithms, which use prior knowledge such as fixed character spacing, connected component analysis and project-based methods to implement segmentation. However, due to the handcraft features often cannot be used to accurately segment, so the segmentation-free algorithm like sequence labelling can effectively avoid the character segmentation error affecting the recognition accuracy. For noisy scenes, some researchers denoise the image and improve the resolution before the license plate detection. For some scenes with skewed shooting angles, some researchers have proposed to use the tilt correction algorithms to correct the license plates or the segmented characters to improve the recognition rate. License plate recognition systems are widely used in modern smart cities, such as toll payment systems, parking fee payment systems and residential access control. Such electronic systems are not only convenient for people's daily life, but also provide safe and efficient services for managers. License plate recognition algorithm is a mature but imperfect technology. The traditional location recognition algorithm is easily affected by light, shadow, background complexity or other factors, resulting in the failure to meet the application of real scenes.

2.3.1 ADVANTAGES

- Present a comprehensive survey on existing license plate systems based on deep learning algorithms, and categorize the algorithms at each stage by the process.

2.3.2 DISADVANTAGES

- Some publicly available license plate datasets are sorted out to compare the amount of each dataset and image resolution.

2.4 TITLE: A Robust License Plate Recognition Model Based on BI-LSTM

AUTHOR: Yongjie Zou

Scene text recognition has been a research interest in the field of computer vision. It is currently using an encoder decoder framework with attention mechanism, to learn the mapping between input images and output sequences in a data-driven manner. License plate recognition can be seen as a special case of general scene text recognition tasks, for they share the similarity of continuous text. This paper proposes a robust model based on license plate recognition in complex scenarios, which is mainly composed of three modules: license plate feature extraction, license plate character positioning, and character feature extraction. The license plate feature extraction module of this model is based on Xception, MobileNetV3, and spatial attention mechanism, which can fully extract the character features of license plates and establish a sound foundation for the license plate character localization. License plate character localization module adopts the method of Bi-LSTM combined with context location information of license plates. From heat maps of license plate positioning in this paper, it can be found that Bi-LSTM can accurately locate the characters of each license plate. Character feature extraction module uses the designed 1D-Attention to enhance the useful character features, and suppress useless character features after Bi-LSTM localization, which plays a key role in character recognition. The accuracy of character recognition directly relates to the quality of character extraction. Through the evaluation of multiple datasets, the experimental results demonstrate the superiority of our proposed method. Whether it is a regular or irregular license plate, or in normal or complex scenes, the method can accurately locate and recognize characters in each license plate.

2.4.1 ADVANTAGES

- Proposes a robust model based on license plate recognition in complex scenarios.

2.4.2 DISADVANTAGES

- Only support Chinese characters categories in the dataset.

2.5 TITLE: Multinational License Plate Recognition Using Generalized Character Sequence Detection

AUTHOR: Chris Henry

Character segmentation is responsible for segmenting individual characters from the detected LP whereas the role of character recognition is to classify each of the segmented characters. The first two steps are crucial for correct ALPR since it directly affects the character recognition stage. Failure to localize the LP in the first stage leads to failure in the subsequent stages. In order to overcome this issue, some literature merges the character segmentation and recognition steps as object recognition steps. Few recent publications proposed end-to-end deep learning structures to completely remove inter-dependency among the three stages. However, as mentioned in the preceding paragraph, these methods are either tailored to work on specific country's LPs or are tested on multi-country datasets that share a common single line LP layout. This paper presents a highly accurate deep ALPR system that is applicable to license plates belonging to multiple countries. In this study, we propose a three-stage deep multinational ALPR approach that combines deep learning with an image processing-based multinational LP layout detection algorithm. LP detection is the first stage of the proposed ALPR system which is responsible for detecting the LP region in an image. This stage uses tiny YOLOv3 network architecture for detecting the LP region and is referred to as the ' attention network' since it provides the LP region to the next stage of our ALPR system. The first two steps are crucial for correct ALPR since it directly affects the character recognition stage. Failure to localize the LP in the first stage leads to failure in the subsequent stages.

2.5.1 ADVANTAGES

- Applicable to license plates from multiple countries by using our proposed multinational LP layout detection algorithm.

2.5.2 DISADVANTAGES

- Proposed algorithms can work on datasets from multiple countries without the need of any additional algorithms or country-specific information.

2.6 TITLE: A Robust Attentional Framework For License Plate Recognition in The Wild

AUTHOR: Linjiang Zhan

This paper tackles the task of license plate recognition in unconstrained scenarios. A robust framework is proposed to handle license plate recognition in both regular and challenging cases effectively. Our proposed license plate recognizer is composed of a 30-layer lightweight Xception for feature extraction and a 2D-attention based decoding module for character sequence recognition. Without extra processings like image rectification or character segmentation, the proposed model is capable of recognizing license plates in both regular and irregular patterns under various practical scenarios. Different from current methods of treating a license plate as a one-dimensional sequence, our method uses 2D-attention that considers license plate image as a 2-dimensional signal. Trained in a weakly supervised manner, the proposed model is able to approximately localize the corresponding characters on license plates in the decoding process, regardless of the appearance of license plate patterns. And design a robust method for license plate recognition in natural scene images. It is made up of a tailored Xception module and an encoder-decoder module. We optimized the recognition framework by using a 2D attention mechanism. It is able to extract local features for individual characters in a weakly supervised manner, without character level annotations needed. Compared to existing license plate recognition approaches, our method does not need an extra module to handle the irregularity of license plates or segment each character for recognition. License plate recognition can be regarded as a special case of general scene text recognition tasks, which have different characteristics. Characters in license plates usually use the same font in one region. There is no language model hidden in the license plate, and no strong relationship with the context semantic information. In contrast, general scene text has a great variability on fonts.

2.6.1 ADVANTAGES

- An LP dataset that contains images captured in different ways from various regions is collected so as to evaluate LP recognition methods more comprehensively.

2.6.2 DISADVANTAGES

- Improving a transformer-like decoder may be explored to accelerate training speed.

2.7 TITLE: License Plate Segmentation and Recognition System Using Deep Learning and Openvino

AUTHOR: Riel D. Castro-Zunt

Automatic License Plate Recognition (ALPR) systems are designed to capture license plates from vehicles and extract identifying information without direct intervention from a human overseer. ALPR is considered a subfield of Intelligent Transportation Systems, services designed to enrich drivers and their interactions on the road. ALPR systems find use in tools for law enforcement: vehicle identification and tracking, issuing fines, restricting area access to certain types of vehicles, and so on. Multiple vehicles in excess of 80 km/h (i.e. highway conditions) may pass a single camera setup at any time. Thus, low processing time and high accuracy is essential to a successful ALPR system. An ALPR system generally performs the following steps sequentially. First, an image is taken. Then, vehicles and/or license plates are detected. Finally, character regions from the license plate are extracted (segmented) and interpreted using an optical character recognition (OCR) method. Oftentimes, license plates within an image are captured blurred, skewed, occluded, with insufficient contrast or brightness, or otherwise unideal. This makes subsequent segmentation and OCR difficult to perform quickly and accurately via purely algorithmic methods. As automated ITS applications become more ubiquitous in society, research into creating fast, accurate, and efficient ALPR solutions becomes necessary. DL, with its demonstrated potential for state-of-the-art computer vision performance, is an excellent candidate for creating ALPR solutions. DL is an application of artificial intelligence with multiple prediction layers to enhance the accuracy of a learned model. DL research is ongoing and cross-discipline, and has demonstrated state-of-the-art performance in various tasks and topics from computer vision to speech recognition.

2.7.1 ADVANTAGES

- Comprised of a feature extractor with DSCs and linear bottlenecks, to preserve accuracy.

2.7.2 DISADVANTAGES

- Future work involves re-training with additional and varied (e.g. non-Californian, non-American) license plate character datasets.

2.8 TITLE: A Robust Deep Learning Approach For Automatic Iranian Vehicle License Plate Detection and Recognition for Surveillance Systems

AUTHOR: Ali Tourani, Asadollah Shabahrami

Nowadays, Intelligent Transportation Systems (ITS) are known as the vital means of municipalities and governments for applying modern traffic planning policies. ITS are mostly integrated with video cameras installed in different locations of the roadways and urban areas for surveillance and enforcement purposes. Since the outputs of these cameras are video frames or a set of images, fetching valuable results from them using machine vision and Digital Image Processing (DIP) techniques, lead to the development of numerous applications for ITS. Among most of these applications, Automatic

License Plate Recognition (ALPR) is considered a major step. ALPR refers to the process of recognizing the characters existing inside the vehicles' license plates through machine vision and artificial intelligence methods. Some of the most common applications in which the license plate recognition process is essential are traffic monitoring systems, navigation and vehicle tracking, toll payments and control systems in public/private parking areas, and the identification of stolen vehicles. In this regard, the correct extraction of characters from license plates is significantly essential to take further actions like enforcement and prosecution. In the character segmentation stage, the system receives the extracted license plates from the LPD stage and produces a set of cropped areas containing the characters. Consequently, the main goal of CS is to detect characters inside the license plate area and segment them from other portions of the license plate, such as a logo or blank fields. Finally, the third phase, i.e., OCR, is responsible for classifying and recognizing each of the segmented characters from the previous stage.

2.8.1 ADVANTAGES

- Vast range of images captured in both challenging and straightforward realistic conditions has been covered in the training process.

2.8.2 DISADVANTAGES

- Increase the time efficiency for each image or video frame to extract the license plate character sequence.

2.9 TITLE: DELP-DAR System for License Plate Detection And Recognition

AUTHOR: Zied Selmia

Automatic detection and recognition of License Plates (LPs) is Frequently used in several areas such as traffic management, digital security surveillance, vehicle recognition, parking management, monitoring border crossings and searching for suspicious vehicles. The process of Automatic LP detection and Recognition (ALPR) is a complex challenge due to many factors that can be divided into two subcategories: technical factors like the various LP numbering systems, types, sizes and languages; and factors related to image quality and environmental conditions such as blurred images, poor lighting conditions, deformation, orientation, different camera angles, daylight, night, raining and complex environmental backgrounds. Several methods and techniques have been used to solve the problems of detecting and recognizing LPs. Among these methods, there have been traditional approaches like edge detection, morphological operations, character-based approaches, texture-based techniques, and statistical analyses. On the other hand, researchers have used learning and classification systems. Recently, the field of deep learning and Convolution Neural Network (CNN) based methods has been used to solve the difficulties found in the detection and recognition of LPs. Despite the use of traditional and deep learning-based methods, several difficulties and challenges are still not properly solved, especially images in complex backgrounds with multi-orientations due to the viewpoint variation in camera and multi-language characters. To resolve the problems mentioned above, we propose here an efficient system for detecting and recognizing LPs. Automatic License Plate detection and Recognition (ALPR) is a quite popular and active research topic in the field of computer vision, image processing and intelligent transport systems. ALPR is used to make detection and recognition processes more robust and efficient in highly complicated environments and backgrounds.

2.9.1 ADVANTAGES

- The increase in the number of characters gives us an improvement in the result.

2.9.2 DISADVANTAGES

- Try to develop a real-time System using new technologies (smartphones, tablets, etc.

2.10 TITLE: Automatic Vehicle License Plate Detection Using K-Means Clustering Algorithm and CNN

AUTHOR: Joy Iong Zong Chen

In recent years, there has been quick progression in Graphical Processing Units (GPU) and Intelligent Transportation System (ITS) and License plate detection and recognition. This has attracted research interests in multiple domains. License Plate Recognition (LPR) has a number of applications such as traffic safety administration, security management and unmanned parking fields. However, because of the distinct format of the plates that vary according to various criteria like distance between vehicle and camera, vehicle speed, brightness, background, and image acquisition. Due to these considerations, there are a number of restrictions like static background, allocated paths, lower vehicle speed and permanent illumination that are required to capture the license plate and identify the characters. There are four common blocks that are used in an LPR technique namely character standardization and classification, segmentation, LP localization and vehicle image. However, because it is tedious to locate the plate, this process is said to be complex and will also have a direct impact on the efficiency and accuracy of the consecutive steps. Hence, it is crucial that the issues due to background disturbances and illumination conditions do not exist. A number of researchers and inventors have proposed many approaches such as line sensitive filters and edge prediction approaches to extract details of arithmetic morphology, window scheme and plate region. Though a number of models proposed so far are capable of determining the position of LP, it has a number of disadvantages such as absence versatility, higher computation time and sensitivity to illumination when used on a diverse platform. By linking components, morphology and relaxation labels, character segmentation is made possible.

2.10.1 ADVANTAGES

- Effectively recognize and detect license plates in the vehicles.

2.10.2 DISADVANTAGES

- Need manual segmentation in number plate detection.

CHAPTER 3

SYSTEM ANALYSIS

3.1 EXISTING SYSTEM

The existing system for managing forest entry access typically involves physical barriers such as gates, guards, and fences, as well as permits or passes that grant access to certain areas of the forest. These methods are effective but require human presence to operate and manage the entry and exit of vehicles.

The physical barrier system involves the installation of gates at the entrance of the forest, which can be opened and closed manually by guards or forest authorities. The gates can be locked during non-operating hours to prevent unauthorized access. While this system is effective in managing access, it requires a significant number of human resources, and there is always the risk of human error.

Template Matching: This algorithm compares a license plate image to a predefined template and searches for similarities between the two images. The algorithm calculates the similarity score between the image and the template, and if the score is above a certain threshold, it recognizes the license plate.

Support Vector Machines (SVM): SVM is a machine learning algorithm that can be used for classification tasks, such as recognizing license plate characters. SVM works by finding the hyperplane that separates different classes of data, and in ANPR systems, it can be used to separate license plate characters from background noise.

3.1.1 LIMITATIONS OF THE EXISTING SYSTEM

- High Computational Requirements: Many of the algorithms used in ANPR systems are computationally intensive, and require significant processing power and memory.
- Limited Accuracy: While ANPR systems have improved in accuracy in recent years, they are still prone to errors, especially in challenging lighting and weather conditions.
- Privacy Concerns: ANPR systems capture and store images of license plates, raising concerns about privacy and surveillance. There are also concerns about the use of ANPR data by law enforcement agencies and other organizations.

3.2 PROPOSED SYSTEM

The proposed system for managing forest entry access involves the integration of NPR and RFID technology, which can provide enhanced security, better management of access permissions, and improved forest management. Under the proposed system, vehicles entering the forest would have their license plate numbers read automatically by an NPR system, which would compare the numbers with a database of registered vehicles.

If the vehicle is registered, the RFID system would automatically grant access without the need for human intervention, allowing the vehicle to pass through gates and other checkpoints quickly and efficiently. Forest authorities can use the system to control the number of vehicles allowed inside the forest at a particular time, reducing overcrowding and minimizing the impact on the forest environment.

The system can also help with tracking vehicles inside the forest and monitoring their movements to prevent illegal activities such as poaching, logging, or dumping. To ensure privacy and data protection, the proposed system would comply with relevant regulations and ensure that the data collected is used for authorized purposes only.

The system would also be designed to minimize the risk of data breaches and cyber-attacks. OCR (Optical Character Recognition) can be used in forest entry access management to read and recognize license plate numbers of vehicles entering or exiting the forest.

The OCR system would use a camera to capture an image of the license plate, which would then be processed by OCR software to extract the license plate number. Finally provide the alert to the user before the exit time and also send notification to admin about exit time in terms of SMS notification.

3.2.1 ADVANTAGES OF THE PROPOSED SYSTEM

Improved Security:

- The ANPR system can help improve security in the forest area by identifying vehicles that are not authorized to be in the area or are involved in suspicious activity. This can help forest authorities to respond quickly to potential security threats.

Traffic Monitoring:

- The ANPR system can be used to monitor traffic in the forest area by capturing and analyzing license plate data. This can help forest authorities to track the movement of vehicles and detect any suspicious activity.

This can help forest authorities to track vehicles that are authorized to be in the forest area and identify any unauthorized vehicles.

3.3 SYSTEM ARCHITECTURE

System architecture refers to the overall design and structure of a computer system or software application. It involves the organization and interconnection of hardware components, software modules, and communication protocols to ensure that the system functions efficiently, reliably, and securely.

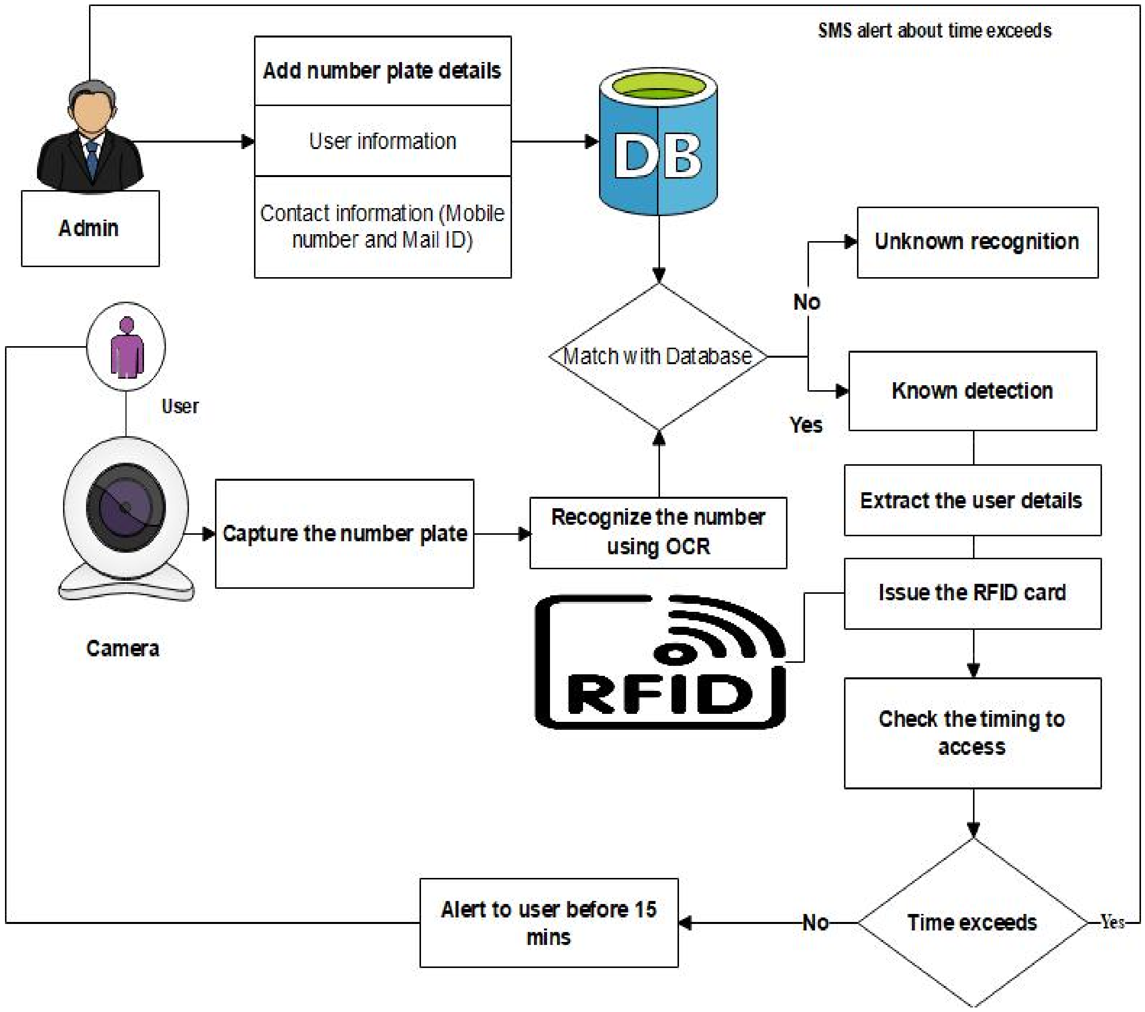


Fig.No: 3.1 SYSTEM ARCHITECTURE

In this diagram, admin can train the user information and contact information with number plate details. In forest entry, capture the number plate to verify the authorized user by using OCR technology. After number plate verification, issue an RFID card to access the forest area. Finally check the notification user about time exceeds and also send notification to admin about time expires.

3.4 MODULES USED

- Framework Construction
- Number Plate Details
- Number Plate Recognition
- RFID Based Access System
- Alert System

3.4.1 FRAMEWORK CONSTRUCTION

A framework is a set of rules, guidelines, and tools that provide structure to a specific development project. In software development, a framework typically includes libraries and APIs that developers can use to build applications, along with recommended architecture and design patterns.

The goal of a framework is to simplify the development process by providing a common set of tools and conventions that can be used across projects, reducing the time and effort required to build a new application from scratch.

In this module, we can design the web application for user and admin. Admin can be responsible for maintaining all details and users can login to the system to post the information.

3.4.2 NUMBER PLATE DETAILS

A number plate, also known as a license plate, is a metal or plastic plate attached to a vehicle that displays its registration number. The registration number is unique to each vehicle and is assigned by a government agency, such as the Department of Motor Vehicles (DMV) in the United States. The purpose of a number plate is to identify the vehicle and its owner for purposes such as road tolls, parking enforcement, and traffic management.

In this we can train the datasets with number plate details. Number Plate details contain information such as user details, address details and so on.

3.4.3 NUMBER PLATE RECOGNITION

Number plate recognition (NPR), also known as license plate recognition (LPR), is a technology that uses optical character recognition (OCR) and computer vision techniques to automatically recognize and extract the characters on a vehicle's license plate.

The extracted information can be used for a variety of purposes, such as automatic toll collection, parking enforcement, and vehicle tracking. The number plate recognition process typically involves the following steps:

- Image acquisition: Capturing an image of a vehicle and its license plate.
- Image pre-processing: Enhancing the image to improve visibility and contrast, and removing noise and artifacts.
- Plate localization: Identifying the region of the image that contains the license plate.
- Character segmentation: Breaking the license plate into individual characters.
- Character recognition: Using OCR to recognize the characters and convert them into text.

In this module, capture the number plate from a forest area. From this data, the number plate can be recognized and matched with the database to extract the owner details.

3.4.4 RFID BASED ACCESS SYSTEM

RFID (Radio-Frequency Identification) based access systems use radio waves to identify and track tags or cards that are attached to objects or carried by people. RFID tags consist of a microchip and an antenna, which work together to transmit and receive data.The reader then checks this code against a database of authorized codes and decides whether or not to grant access to the person or object carrying the tag.

In this module, authorities issue the RFID card to users to visit the forest area. Each individual has a unique RFID number and is stored in a database.

3.4.5 ALERT SYSTEM

In this module, recognized user details are extracted from databases which are extracted from trained databases. CNNs will compare input images pixel by pixel or group of boxes. The regions that appear are called landscapes.

By definition of rough feature contests in roughly the similar places in two images, convolutional neural networks get a lot of improvement at sighted likeness than entire–image matching patterns. And send the fine amount details to the appropriate user in the form of an SMS alert.

CHAPTER 4

SYSTEM DESIGN

4.1 UML DIAGRAMS

A Unified Modeling Language (UML) diagram provides a visual representation of an aspect of a system. UML diagrams illustrate the quantifiable aspects of a system that can be described visually, such as relationships, behavior, structure, and functionality.

For example, a class diagram describes the structure of the system or the details of an implementation, while a sequence diagram shows the interaction between objects over time.

In a UML diagram, the diagram elements visually represent the classifiers in a system or application. These classifiers are the diagrammatic representation of a source element. UML diagrams provide views of source elements; however, diagram elements do not have semantic value.

UML diagrams can help system architects and developers understand, collaborate on, and develop an application. High-level architects and managers can use UML diagrams to visualize an entire system or project and separate applications into smaller components for development.

System developers can use UML diagrams to specify, visualize, and document applications, which can increase efficiency and improve their application design. UML diagrams can also help identify patterns of behavior, which can provide opportunities for reuse and streamlined applications. The visual representation of a system that UML diagrams provide can offer both low-level and high-level insight into the concept and design of an application.

4.2 TYPES OF UML DIAGRAMS

- Use Case Diagram
- Class Diagram
- Sequence Diagram
- Collaboration Diagram
- Activity Diagram

4.2.1 USE CASE DIAGRAM

A Use Case is a list of steps, typically defining interactions between a role (known in Unified Modeling Language (UML) as an "actor") and a system, to achieve a goal.

The actor can be a human, an external system, or time. In systems engineering, use cases are used at a higher level than within software engineering, often representing missions or stakeholder goals.

Use Case Diagram has actors like admin and user. Use cases show the activities handled by both sender and receiver. Figure 4.1 shows the use case diagram for the proposed system.

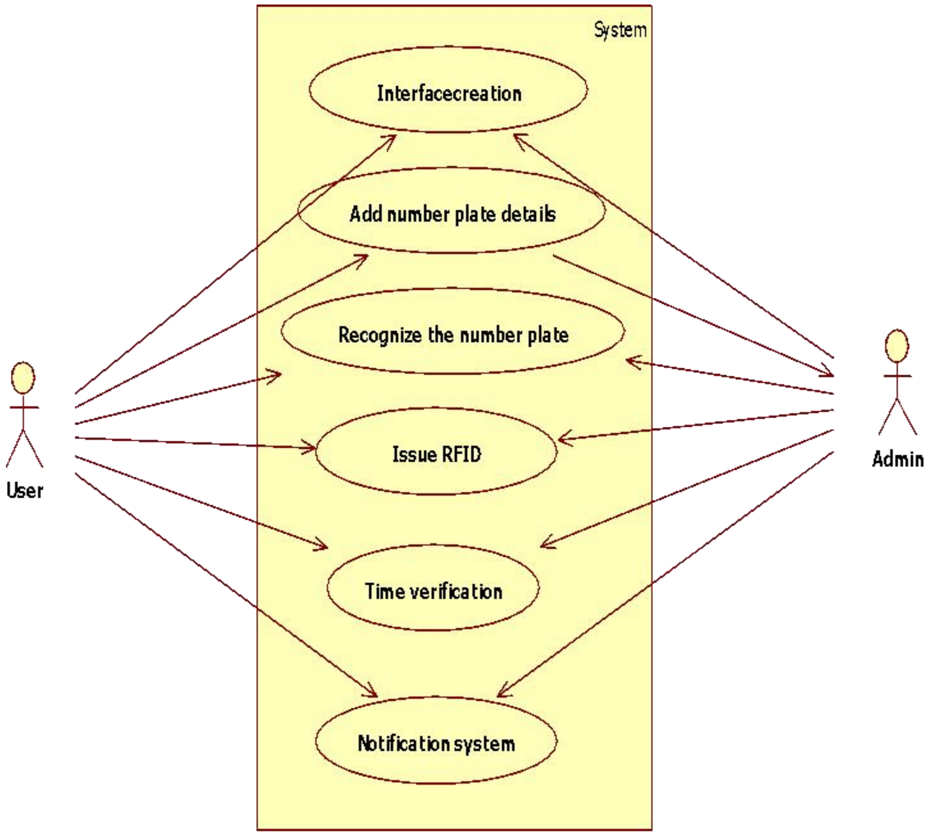


Fig.No: 4.1 USE CASE DIAGRAM

4.2.2 CLASS DIAGRAM

A Class diagram is the main building block of object-oriented modeling. It is used for general conceptual modeling of the structure of the application and for detailed modelling translating the models into programming code. Class diagrams can also be used for data modelling.

The classes in a class diagram represent both the main elements, interactions in the application and the classes to be programmed. Figure 4.2 shows the class diagram for the proposed system.

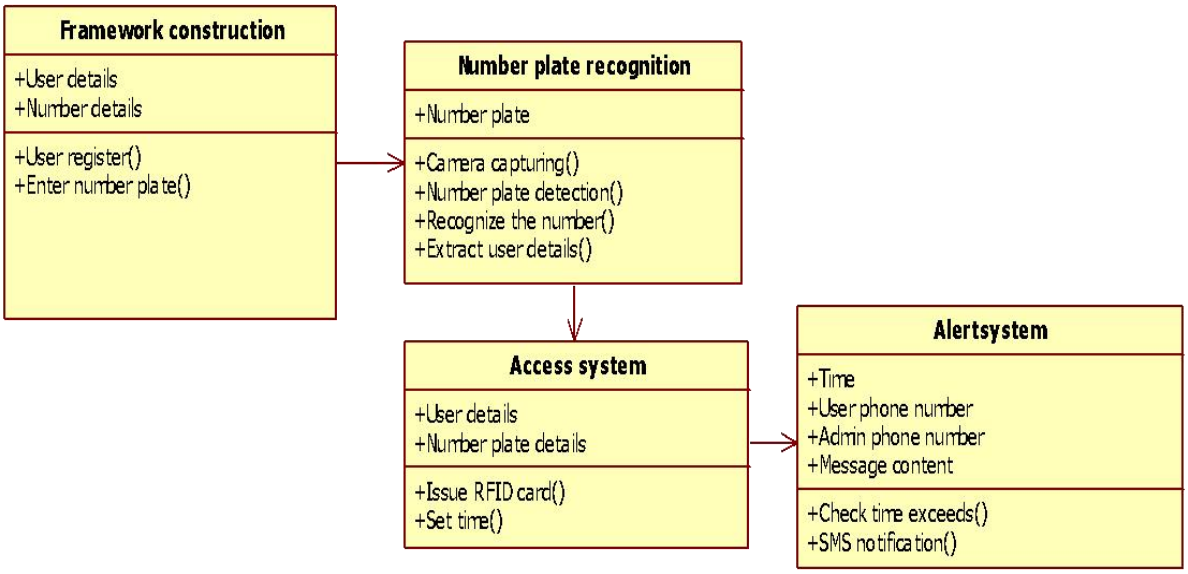


Fig. No: 4.2 CLASS DIAGRAM

4.2.3 SEQUENCE DIAGRAM

A Sequence diagram is an interaction diagram that shows how processes operate with one another and in what order. It is a construct of a Message Sequence Chart. A sequence diagram shows object interactions arranged in time sequence.

Sequence diagrams are sometimes called event trace diagrams, event scenarios and timing diagrams. A sequence diagram shows, as parallel vertical lines, different processes that live simultaneously and horizontal arrows. The messages exchanged between them.

Sequence diagram has three objects. The connection between the objects is mentioned using stimulus and self-stimulus. Figure 4.3 shows the sequence diagram for the proposed system.

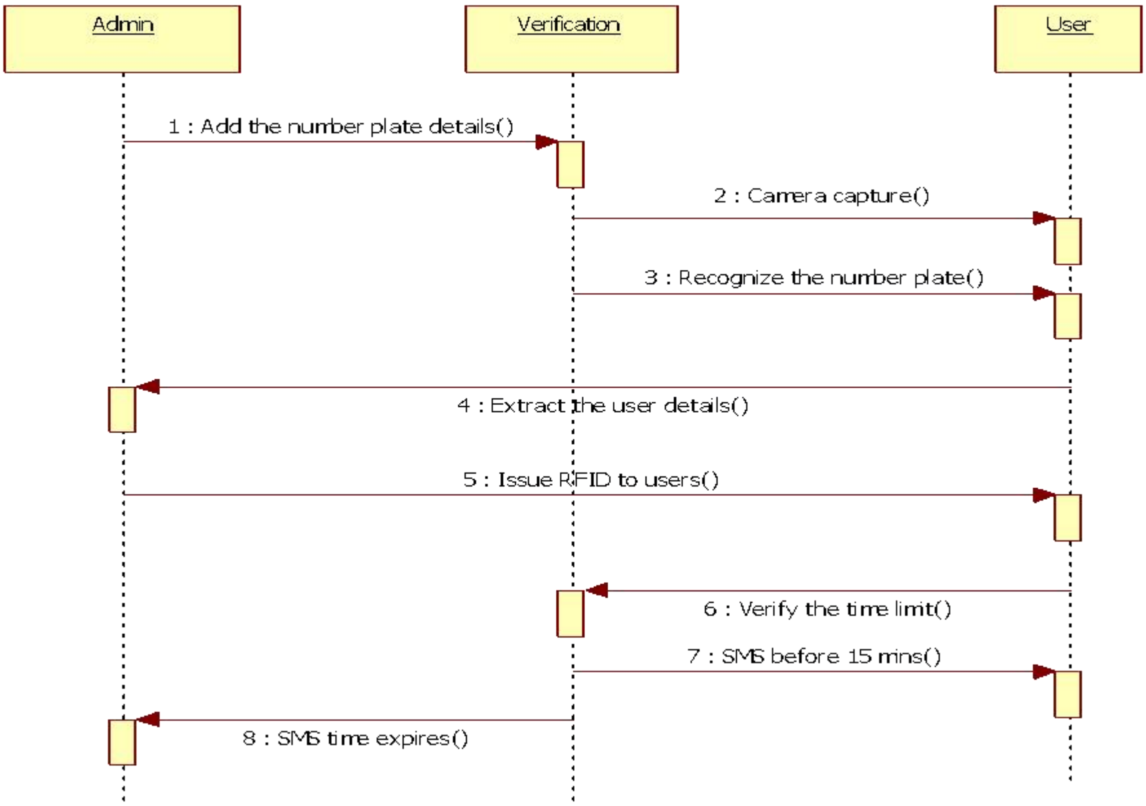


Fig .No 4.3 SEQUENCE DIAGRAM

4.2.4 COLLABORATION DIAGRAM

A collaboration diagram resembles a flowchart that portrays the roles, functionality and behavior of individual objects as well as the overall operation of the system in real time. Objects are shown as rectangles with naming labels inside. These labels are preceded by colons and may be underlined. The relationships between the objects are shown as lines connecting the rectangles.

The messages between objects are shown as arrows connecting the relevant rectangles along with labels that define the message sequencing. Collaboration diagrams show the collaborative connections between three objects like sender, receiver and server. Collaborative has self stimulus and also connection between two objects. Figure 4.4 shows the collaboration diagram for the proposed system.

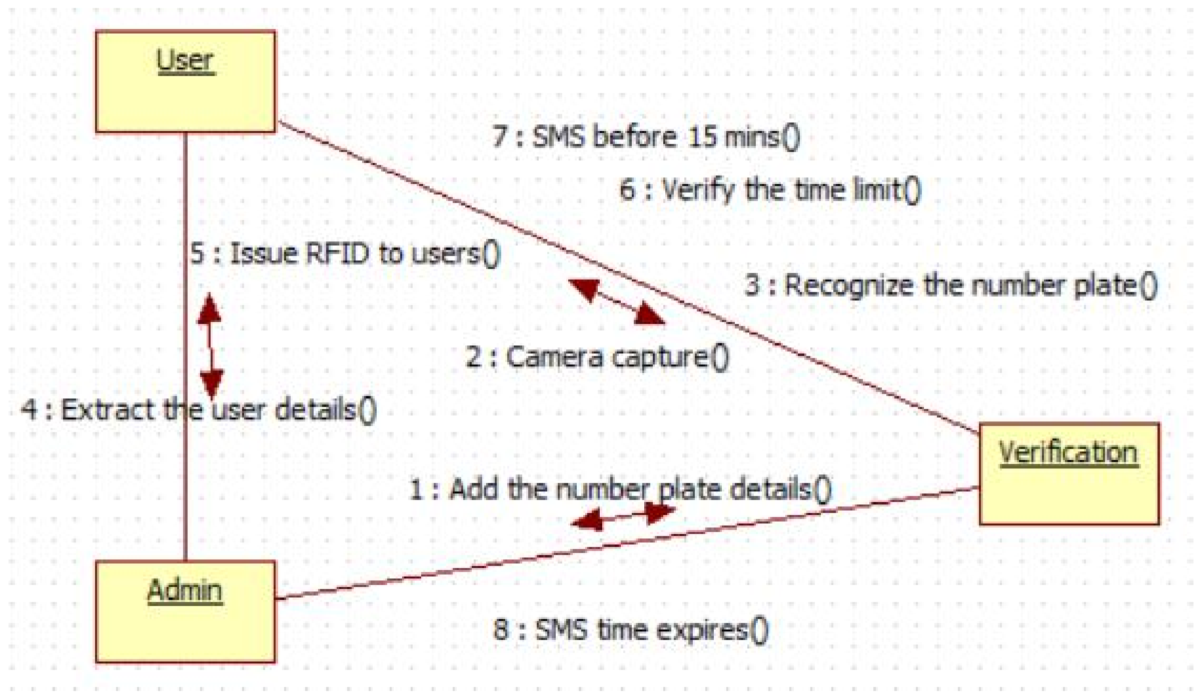


Fig .No: 4.4 COLLABORATION DIAGRAM

4.2.5 ACTIVITY DIAGRAM

Activity diagrams are graphical representations of workflows of stepwise activities and action with support for choice, iteration and concurrency. In the Unified Modeling Language, activity diagrams are intended to model both computational and organizational processes.

Activity diagrams show the overall flow of control. Activity diagram has an initial and final state. Then activities are mentioned between the states. Figure 4.5 shows the activity diagram for the proposed system.

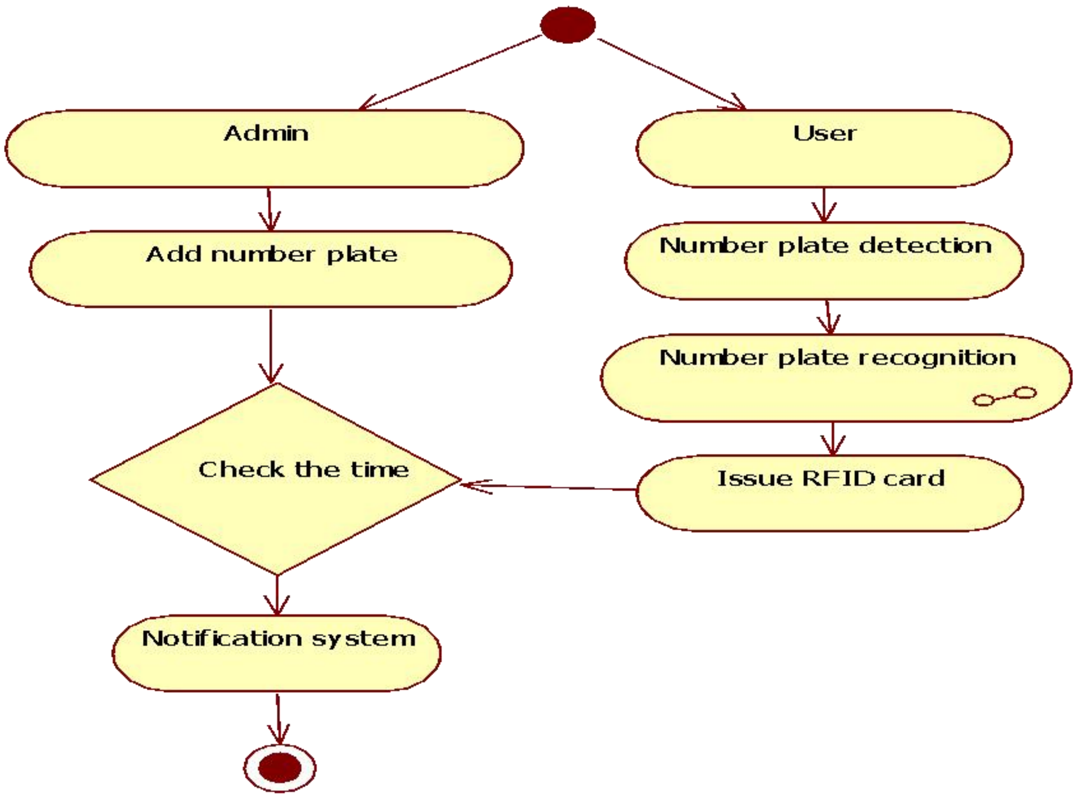


Fig. No: 4.5 ACTIVITY DIAGRAM

CHAPTER 5

5. IMPLEMENTATION

5.1 NUMBER PLATE RECOGNITION – OPTICAL CHARACTER RECOGNITION

OCR (Optical Character Recognition) is an essential technology used in number plate recognition. It enables cameras to capture an image of a vehicle's license plate and converts the image into text data that can be recognized and processed by a computer. OCR-based number plate recognition systems work by analyzing the captured image of a vehicle's license plate to identify the characters on the plate.

This involves several stages of processing, including image enhancement, character segmentation, character recognition, and verification. During the image enhancement stage, the image is processed to remove any noise, blur, or other distortions that may affect the accuracy of the recognition process. This is followed by character segmentation, where the characters on the plate are identified and separated from each other.

Once the characters are segmented, they are recognized using OCR software. OCR technology is capable of recognizing characters in different fonts, sizes, and styles, making it a reliable method for recognizing license plate numbers. After the characters are recognized, they are verified to ensure that the correct characters have been identified.

The verification process may involve checking the recognized characters against a database of known license plate numbers or performing a checksum calculation to verify the validity of the plate number. OCR-based number plate recognition systems offer several benefits, including increased accuracy and speed compared to manual recognition methods. The OCR (Optical Character Recognition) algorithm consists of several steps to recognize characters accurately. The following are the typical steps involved in the OCR algorithm:

Image acquisition: The first step in the OCR algorithm is to acquire an image of the text to be recognized. The image can be obtained using a scanner or a camera.

Preprocessing: Once the image is acquired, it is preprocessed to remove any noise or distortions that may affect the recognition process. This step involves several processes, such as image enhancement, noise reduction, and contrast adjustment.

Binarization: After preprocessing, the image is converted into a binary format where the background is set to black, and the text is set to white. This step helps in separating the text from the background.

Segmentation: The next step is to segment the image into individual characters or words. This is done by identifying the boundaries between characters or words.

Feature extraction: Once the characters are segmented, features such as size, shape, and texture are extracted to distinguish between different characters. This step helps in improving the accuracy of character recognition.

Character recognition: In this step, the features extracted from the segmented characters are compared with a pre-existing database of characters to recognize the characters accurately.

Post-processing: The final step in the OCR algorithm is post-processing. In this step, the recognized characters are verified for errors and corrected if necessary.

Overall, the OCR algorithm is a complex process that involves several steps to recognize characters accurately. While the OCR algorithm can achieve high accuracy levels, it is essential to preprocess the image correctly and extract the right features to improve the recognition accuracy.

PSEUDOCODE

```
function OCR(image):
    preprocessed_image = preprocess(image)
    character_segments = segment(preprocessed_image)
    recognized_text = ""
    for each character_segment in character_segments:
        character_features = extract_features(character_segment)
        recognized_character = recognize_character(character_features)
        recognized_text = recognized_text + recognized_character
        return recognized_text
```

5.3 SYSTEM SPECIFICATION

5.3.1 HARDWARE REQUIREMENTS

- Processor : Intel core processor 2.6.0 GHZ
- RAM : 1GB
- Hard disk : 160 GB
- Compact Disk : 650 Mb
- Keyboard : Standard keyboard
- Monitor : 15-inch color monitor

5.3.2 SOFTWARE REQUIREMENTS

- Operating system : Windows OS
- Front End : PYTHON
- Back End : MYSQL
- IDE : PYCHARM

CHAPTER 6

6. CONCLUSION AND FUTURE WORK

6.1 RESULT AND DISCUSSION

The results of a number plate recognition system depend on various factors such as the accuracy of the OCR algorithm, the quality of the input image, and the effectiveness of the segmentation and feature extraction techniques. If the OCR algorithm is accurate, the system can recognize the characters on the number plate correctly, which can improve the overall efficiency of the system.

Additionally, if the segmentation and feature extraction techniques are effective, the system can separate the characters from the background and extract features that can be used to recognize the characters accurately. One of the common metrics used to evaluate the accuracy of the number plate recognition system is the recognition rate, which measures the percentage of correctly recognized characters. The recognition rate can be affected by factors such as image resolution, noise, and lighting conditions. In some cases, the number plate recognition system may not be able to recognize the characters correctly due to factors such as image quality or the presence of special characters or symbols that are not present in the pre-existing database of characters.

In such cases, the system may need to be adjusted to improve the recognition accuracy. Overall, the results and discussion of the number plate recognition system depend on various factors, and it is essential to evaluate the system's accuracy under various conditions to ensure its effectiveness. In addition to the accuracy of the OCR algorithm, there are other factors that can affect the performance of a number plate recognition system. One of the critical factors is the quality of the input image. The image must be clear and well-focused, with sufficient contrast and brightness to make it easy to recognize the characters.

6.2 CONCLUSION

In conclusion, implementing a number plate recognition system with RFID technology in forest areas can provide various benefits for forest management, such as improved access control, enhanced resource allocation, and improved safety and security. Therefore, it is essential to consider implementing such a system in forest areas to enhance forest management effectiveness.

By integrating RFID technology, the system can ensure that only authorized vehicles are allowed entry into the forest, and the number plate recognition system can help identify vehicles that violate forest regulations. The number plate recognition system can recognize the characters on the number plate accurately, provided that the input image is of sufficient quality and the OCR algorithm is effective.

The system's performance can be further improved by using effective segmentation. However, the success of the system depends on various factors such as the quality of the RFID tags, the reliability of the communication network. Therefore, it is essential to consider these factors while implementing the system to ensure its effectiveness.

6.2 FUTURE WORK

The system could be integrated with other technologies such as GPS tracking, drones, and sensors to provide a more comprehensive and real-time monitoring of forest activities. This could help to identify potential risks such as forest fires, illegal logging, and poaching, and enable quick and effective response.

APPENDICES

APPENDIX I

SOURCE CODE

```
from flask import Flask, render_template, flash, request, session
from flask import render_template, redirect, url_for, request
import mysql.connector
import datetime
import time
app = Flask(__name__)
app.config['DEBUG']
app.config['SECRET_KEY'] = '7d441f27d441f27567d441f2b6176a'

@app.route("/")
def homepage():
return render_template('index.html')

@app.route("/AdminLogin")
def AdminLogin():
return render_template('AdminLogin.html')

@app.route("/UserLogin") def
UserLogin():
```

```
return render_template('UserLogin.html')

@app.route("/NewUser")
def NewUser():
    return render_template('NewUser.html')

@app.route("/AdminHome")
def AdminHome():
    conn = mysql.connector.connect(user='root', password='', host='localhost',
database='2forestentrydb')
    cur = conn.cursor()
    cur.execute("SELECT * FROM regtb ")
    data = cur.fetchall()
    return render_template('AdminHome.html', data=data)


@app.route("/OutEntry")
def OutEntry():
    conn = mysql.connector.connect(user='root', password='', host='localhost',
database='2forestentrydb')
    cur = conn.cursor()
    cur.execute("SELECT * FROM entrytb where Status='in'")
    data = cur.fetchall()

    return render_template('OutEntry.html', data=data)
```

```
@app.route("/AdminReport")
def AdminReport():
    from datetime import datetime
    import pandas as pd
    # Get current time in local timezone
    current_time = datetime.now()
    print('Current timestamp: ', current_time.strftime('%H:%M:%S'))

    dd1 = current_time.strftime('%H:%M:%S')

    n = 15
    # Add 2 minutes to datetime object containing current time
    future_time = current_time + pd.DateOffset(minutes=n)
    print('Future Time (2 minutes from now ): ', future_time)
    # Convert datetime object to string in specific format
    future_time_str = future_time.strftime('%H:%M:%S')
    print('Future Time as string object: ', future_time_str)
    dd2 = future_time_str
    conn = mysql.connector.connect(user='root', password='', host='localhost',
database='2forestentrydb')
    cur = conn.cursor()
    cur.execute("SELECT * FROM entrytb where OutTime between '" + dd1 +
"' and '" + dd2 + "' and Status='in' ")
    data = cur.fetchall()
    for item1 in data:
```

```
        uname = item1[1]
        outtime = item1[5]

        conn = mysql.connector.connect(user='root', password='', host='localhost',
database='2forestentrydb')
        cur = conn.cursor()
        cur.execute("SELECT * FROM regtb where Username='" + uname + "'")
        data2 = cur.fetchall()
        for item in data2:
            email = item[2]
            sendmsg(email, "Out Time" + str(outtime))

    conn = mysql.connector.connect(user='root', password='', host='localhost',
database='2forestentrydb')
    cur = conn.cursor()
    cur.execute("SELECT * FROM entrytb where OutTime <= '" + dd1 + "' and
Status='in' ")
    data = cur.fetchall()
    for item1 in data:
        uname = item1[2]

        sendmsg("9384410144", "Not Exist, vehicle No" + uname)

    conn = mysql.connector.connect(user='root', password='', host='localhost',
database='2forestentrydb')
    cur = conn.cursor()
    cur.execute("SELECT * FROM entrytb ")
    data = cur.fetchall()
    return render_template('AdminReport.html', data=data)
```

```
@app.route("/adminlogin", methods=['GET', 'POST'])
def adminlogin():
   error = None
   if request.method == 'POST':
      if request.form['Name'] == 'admin' and request.form['Password'] == 
'admin':
         conn = mysql.connector.connect(user='root', password='', 
host='localhost', database='2forestentrydb')
         cur = conn.cursor()
         cur.execute("SELECT * FROM regtb ")
         data = cur.fetchall()
         return render_template('AdminHome.html', data=data)
      else:
         data = "UserName or Password Incorrect!"
         return render_template('goback.html', data=data)
@app.route("/newuser", methods=['GET', 'POST'])
def newuser():
   if request.method == 'POST':
      name = request.form['t1']
```

```
mobile  =  request.form['t2']
email   =  request.form['t3']
vno     =  request.form['t6']
username = request.form['t4']
Password = request.form['t5']
vtype = request.form['vtype']

conn = mysql.connector.connect(user='root', password='', host='localhost',
database='2forestentrydb')
cursor = conn.cursor()
cursor.execute("SELECT * from regtb where username='" + username + "'
or  VehicleNo='" + vno + "'")
data = cursor.fetchone()
if data:
   data = "Already Register  VehicleNo Or UserName!"
   return render_template('goback.html', data=data)

else:
   conn = mysql.connector.connect(user='root', password='',
host='localhost', database='2forestentrydb')
   cursor = conn.cursor()
   cursor.execute(
      "insert into regtb values('','" + name + "','" + mobile + "','" + email +
"','" + vno + "','" + username + "','" + Password + "','" + vtype + "')")
   conn.commit()
   conn.close()
   data = "Record Saved!"
   return render_template('goback.html', data=data)
```

```
@app.route("/userlogin", methods=['GET', 'POST'])
def userlogin():
   if request.method == 'POST':
      username = request.form['Name']
      password = request.form['Password']
      # session['uname'] = request.form['uname']

      conn = mysql.connector.connect(user='root', password='', host='localhost',
database='2forestentrydb')
      cursor = conn.cursor()
      cursor.execute("SELECT * from regtb where username='" + username + "'
and Password='" + password + "'")
      data = cursor.fetchone()
      if data is None:
         alert = 'Username or Password is wrong'
         return render_template('goback.html', data=alert)

      else:
         session['vno'] = data[4]
         session['uname'] = data[5]
         session['mob'] = data[2]
```

```
        conn = mysql.connector.connect(user='root', password='',
host='localhost', database='2forestentrydb')
        # cursor = conn.cursor()
        cur = conn.cursor()
        cur.execute("SELECT * FROM regtb where username='" + username +
"' and Password='" + password + "'")
        data = cur.fetchall()

        return render_template('UserHome.html', data=data)

@app.route("/UserHome")
def UserHome():
   username = session['uname']


   conn = mysql.connector.connect(user='root', password='', host='localhost',
database='2forestentrydb')
   cur = conn.cursor()
   cur.execute("SELECT * FROM regtb  where username='" + username + "' ")
   data = cur.fetchall()
   return render_template('UserHome.html', data=data)
```

```
    cur.execute("SELECT * FROM  entrytb  where UserName='" + uname + "'
")
    data = cur.fetchall()
    return render_template('EntryInfo.html', data=data)

import cv2
import numpy as np
from skimage.filters import threshold_local
import tensorflow as tf
from  skimage  import  measure
import imutils
import pytesseract
import re
import mysql.connector

def     sort_cont(character_contours):
    """
    To sort contours from left to right
    """
    i = 0
    boundingBoxes  =  [cv2.boundingRect(c)  for  c  in  character_contours]
    (character_contours, boundingBoxes) = zip(*sorted(zip(character_contours,
boundingBoxes),
key=lambda b: b[1][i], reverse=False))
    return character_contours

        ars(plate, 400)if charactersFound:
            return charactersFound


    # PLATE FEATURES
```

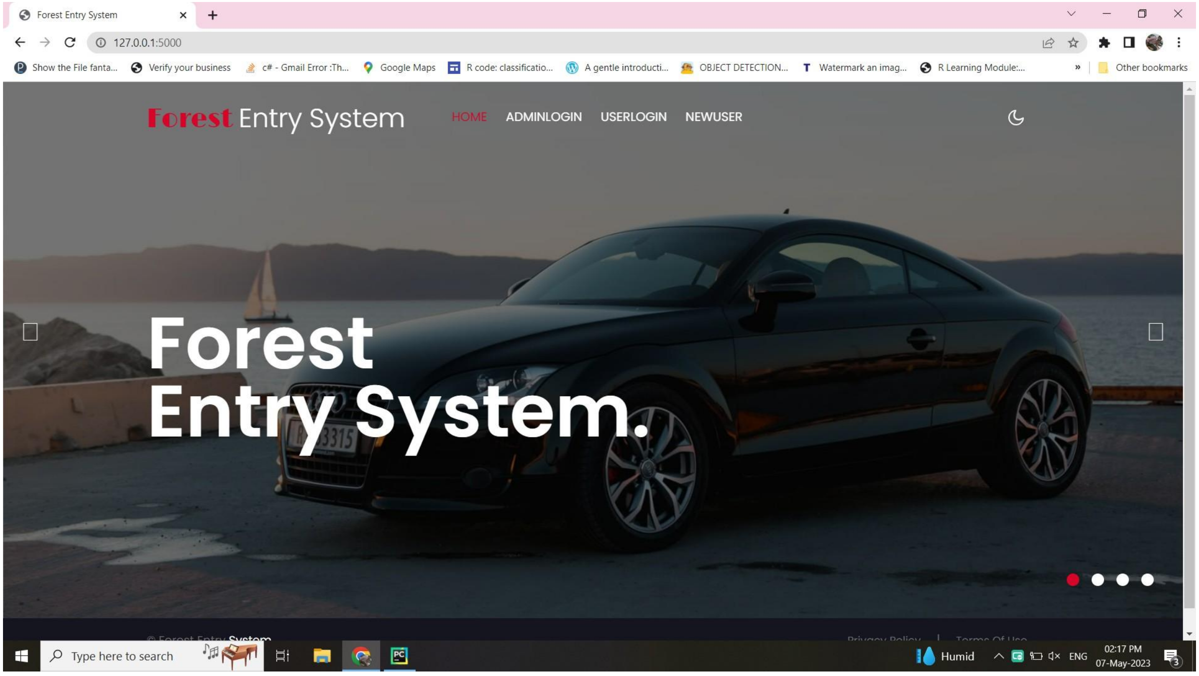
```
def ratioCheck(self, area, width, height):
```

```
        "update  entrytb set Status='out' where Rfid='" + rfid + "' and Date='"+
date +"' and Status='in' ")
        conn.commit()
        conn.close()

        data = "Exit info Update!"
        return render_template('goback.html', data=data)

if__name__== '__main__':
    app.run(debug=True, use_reloader=True)
```

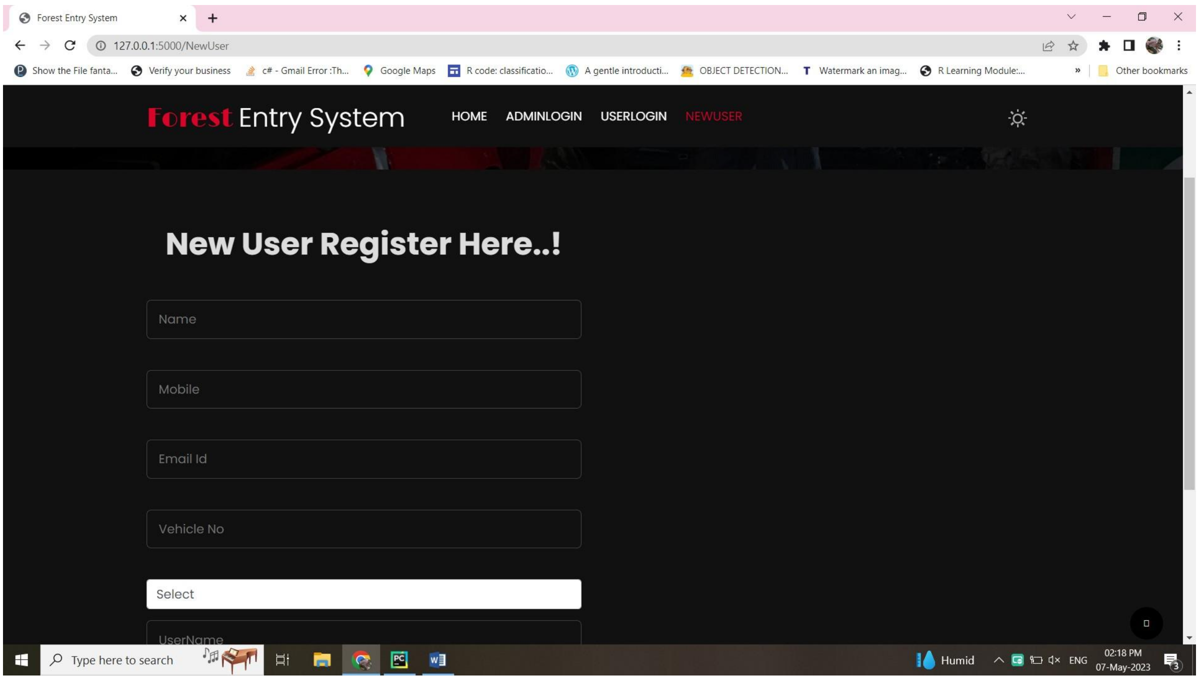
APPENDIX II

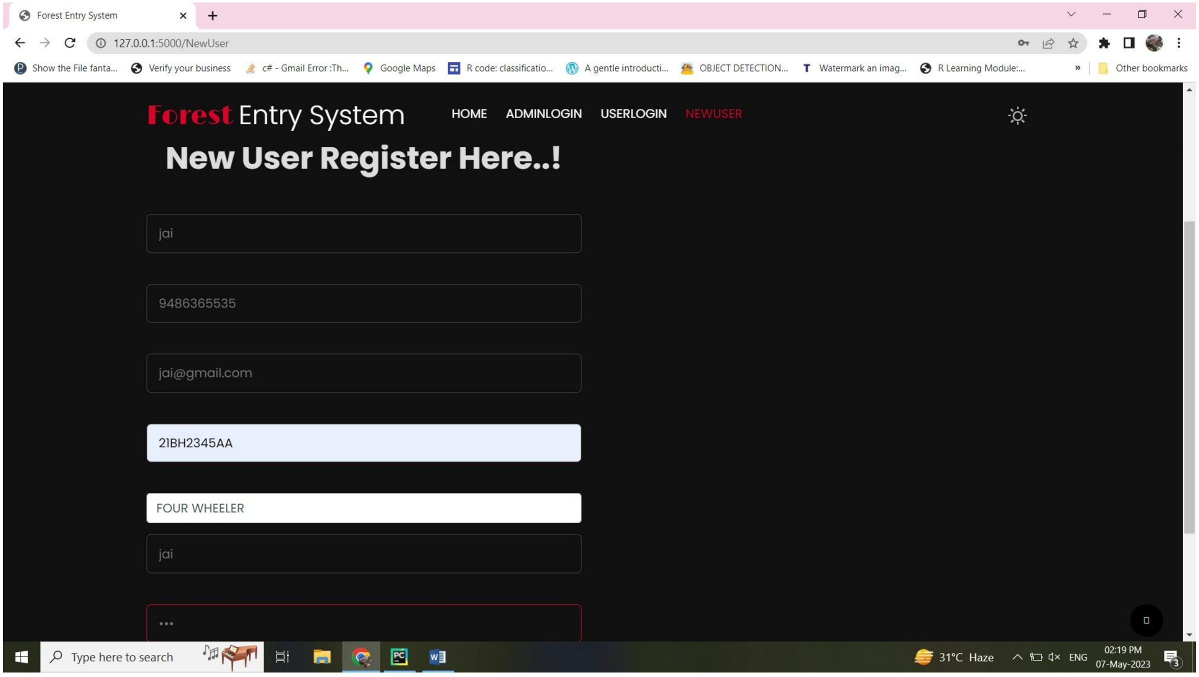
SCREENSHOTS

HOMEPAGE

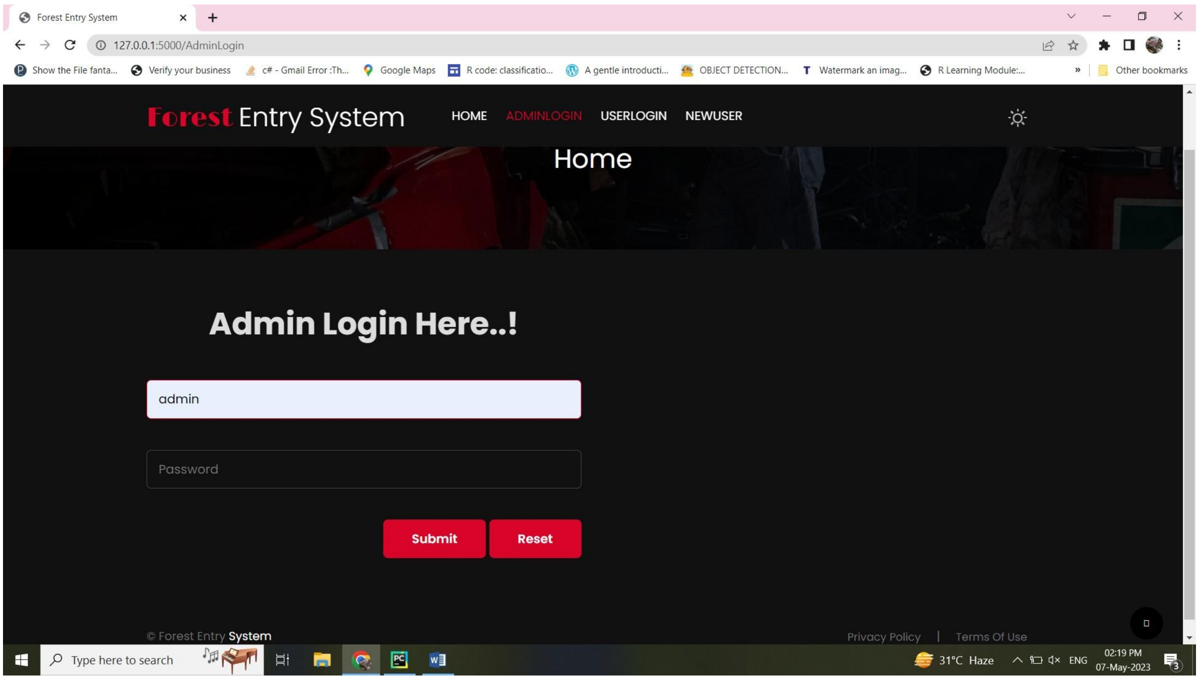


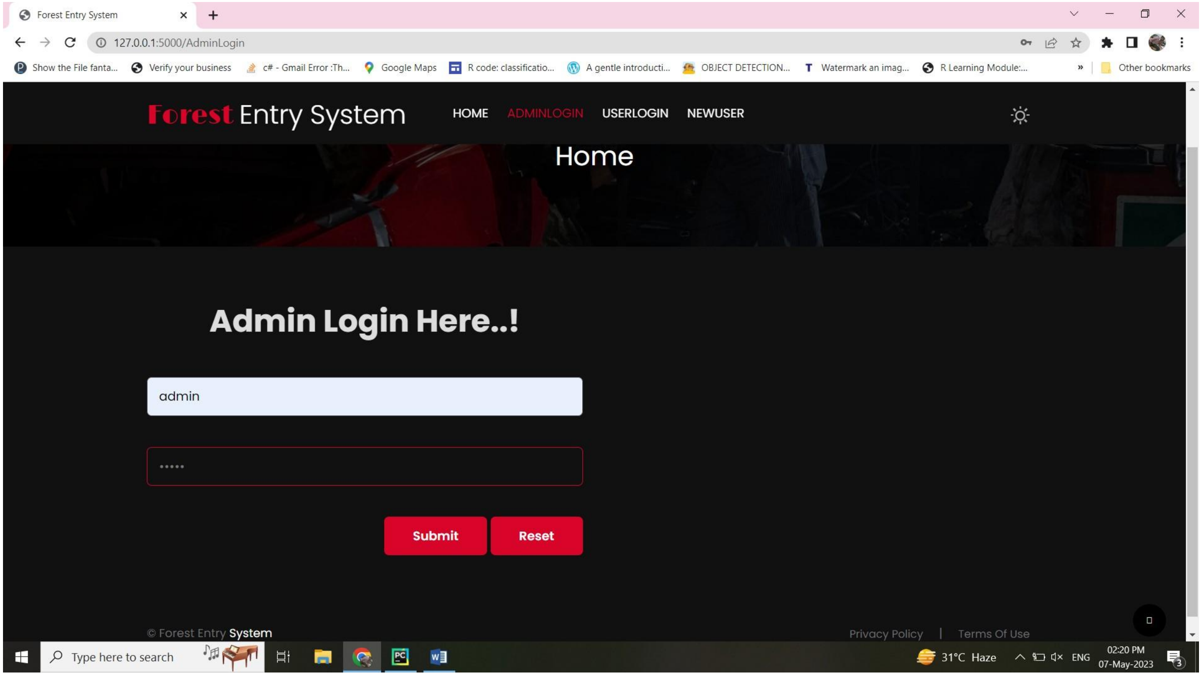
NEW USER REGISTRATION



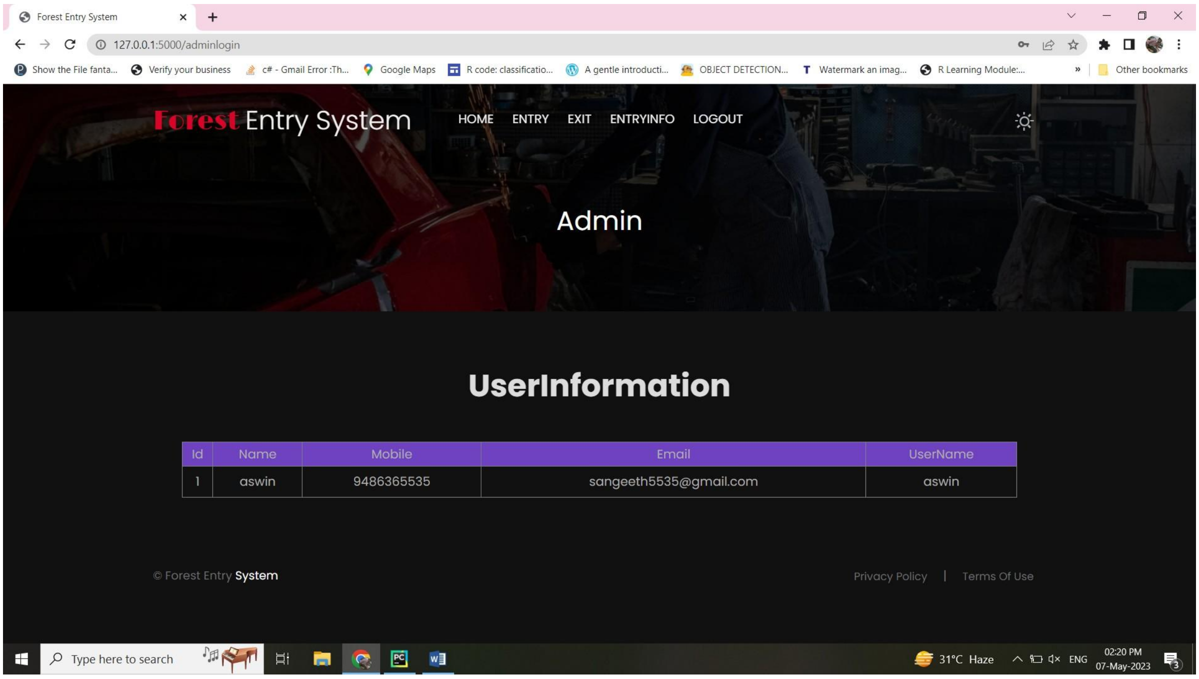

Forest Entry System
127.0.0.1:5000/NewUser
Forest Entry System
HOME
ADMINLOGIN
USERLOGIN
NEWUSER
New User Register Here..!
jai
9486365535
jai@gmail.com
21BH2345AA
FOUR WHEELER
jai
Type here to search
31°C Haze
ENG
02:19 PM
07-May-2023

ADMIN LOGIN

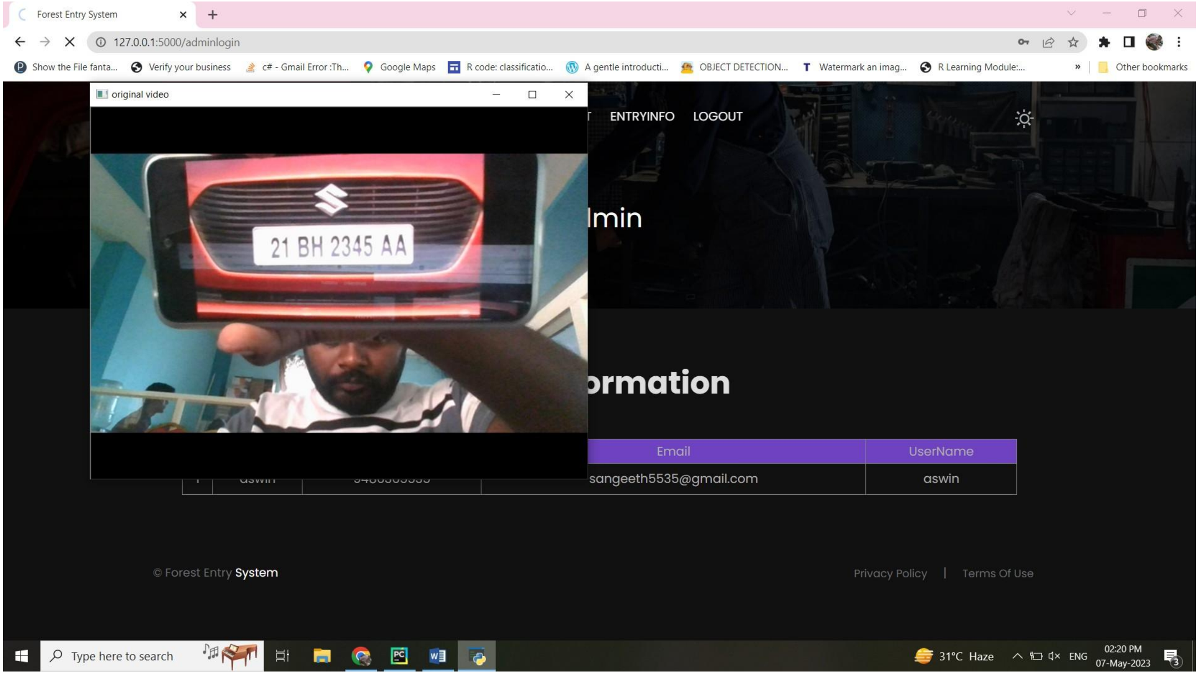

Forest Entry System
127.0.0.1:5000/AdminLogin
Forest Entry System
HOME
ADMINLOGIN
USERLOGIN
NEWUSER
Home
Admin Login Here..!
admin
Password
Submit
Reset
© Forest Entry System
Privacy Policy
Terms Of Use
Type here to search
31°C Haze
ENG
02:19 PM
07-May-2023



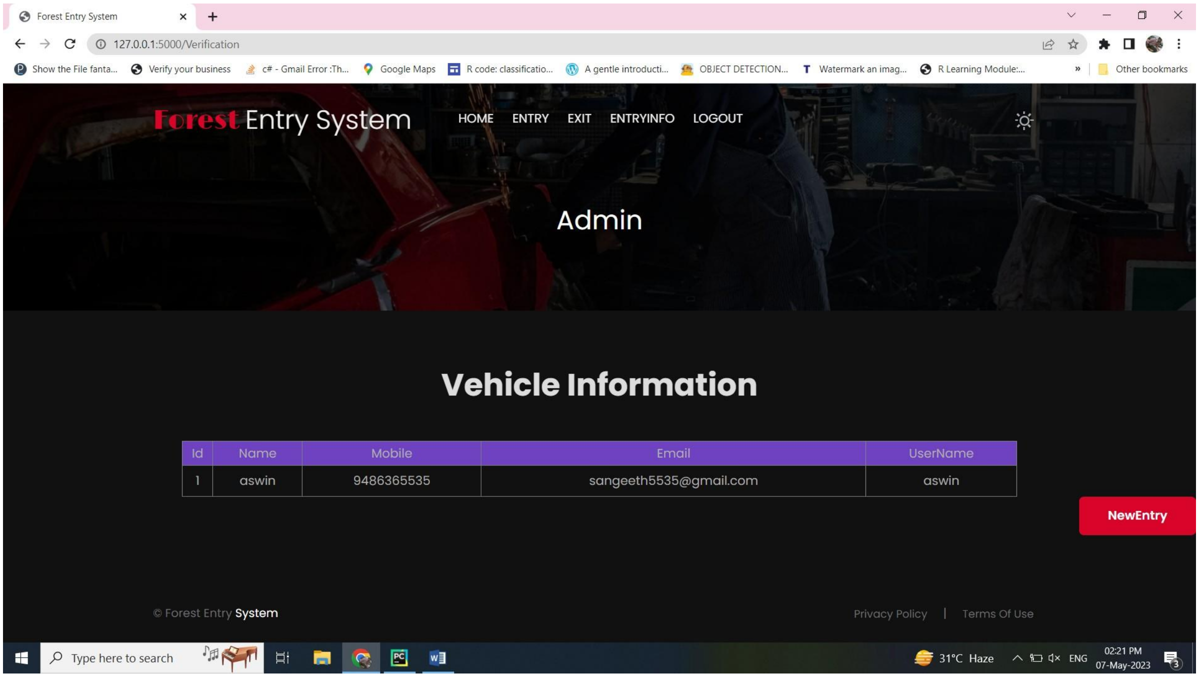
USER DETAILS



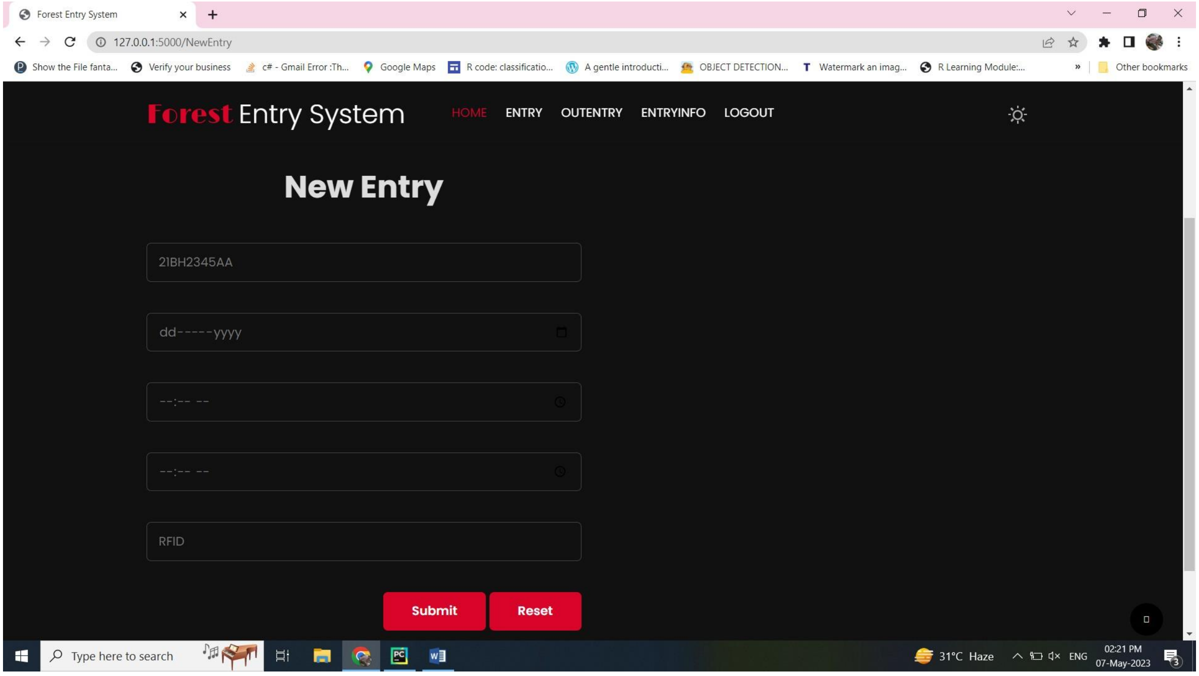
NUMBER PLATE RECOGNITION

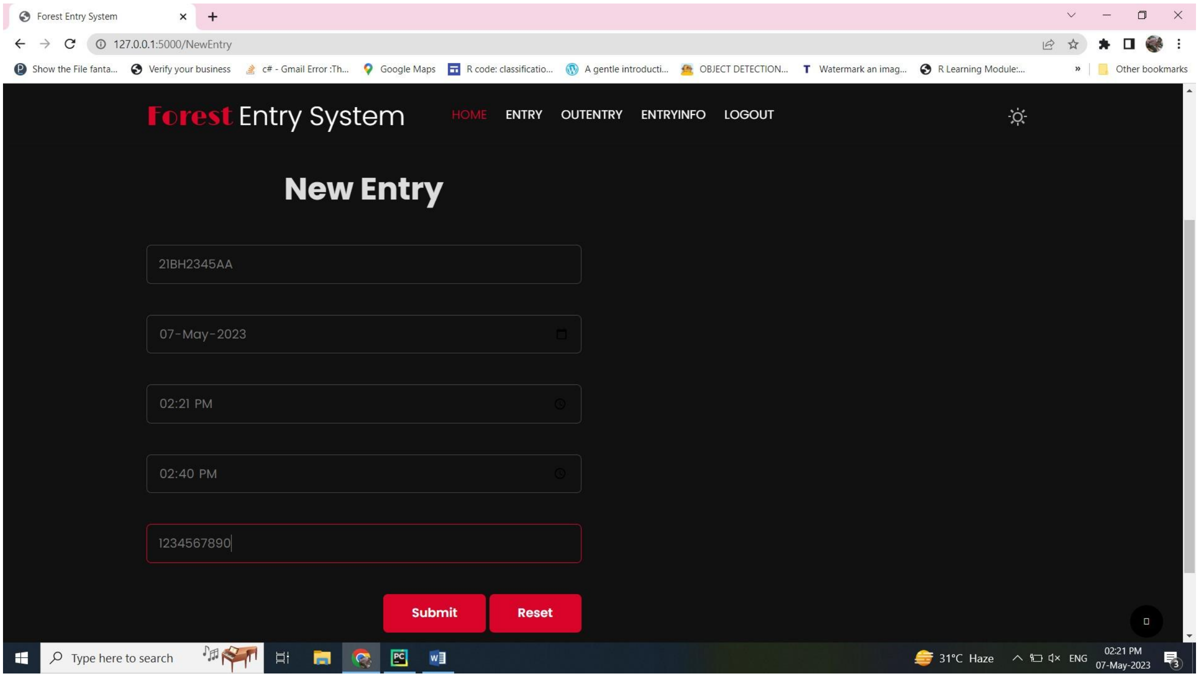


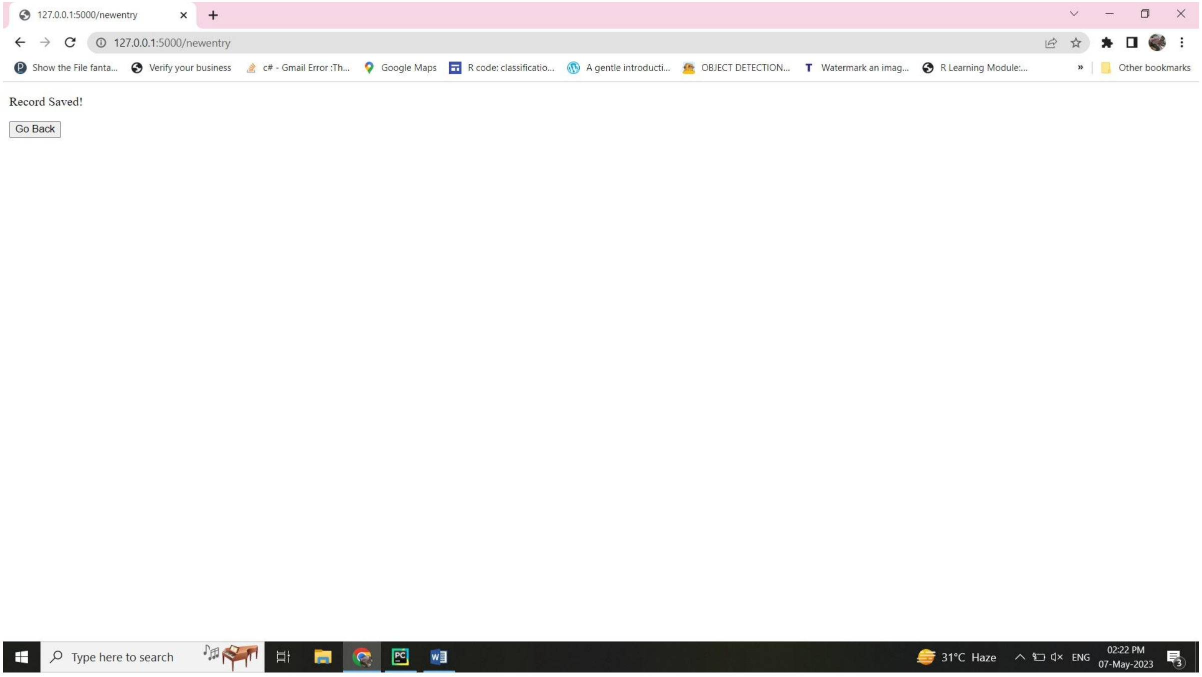
VEHICLE INFORMATION



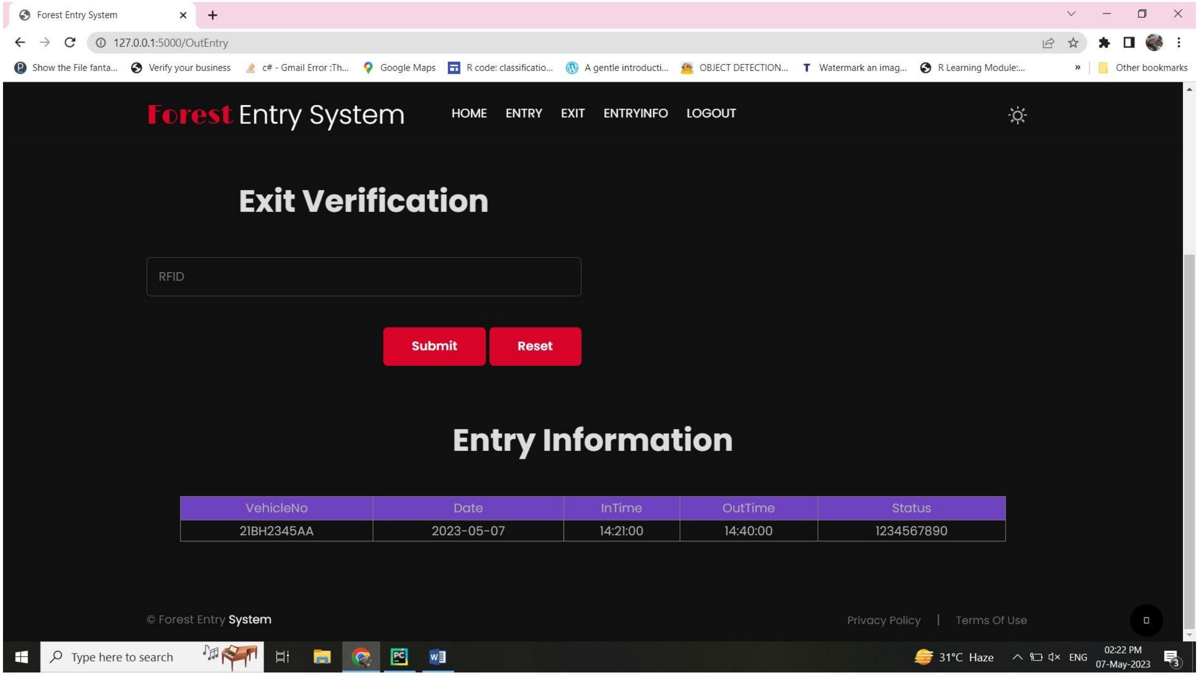
VEHICLE ENTRY FORM

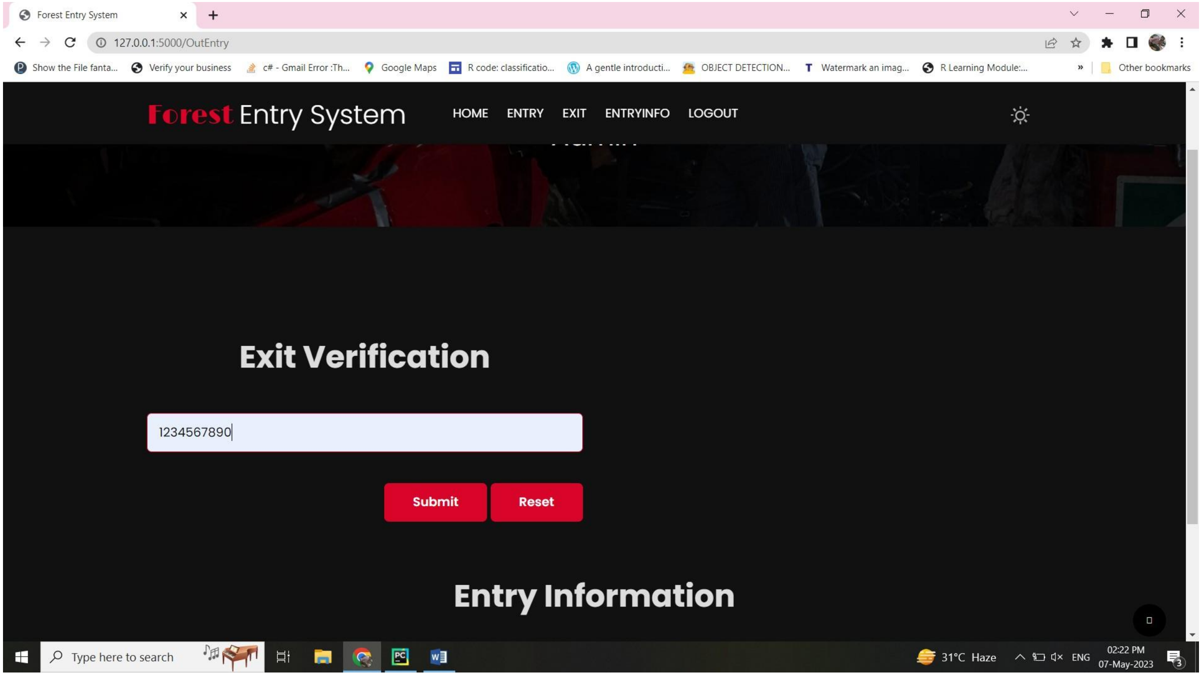


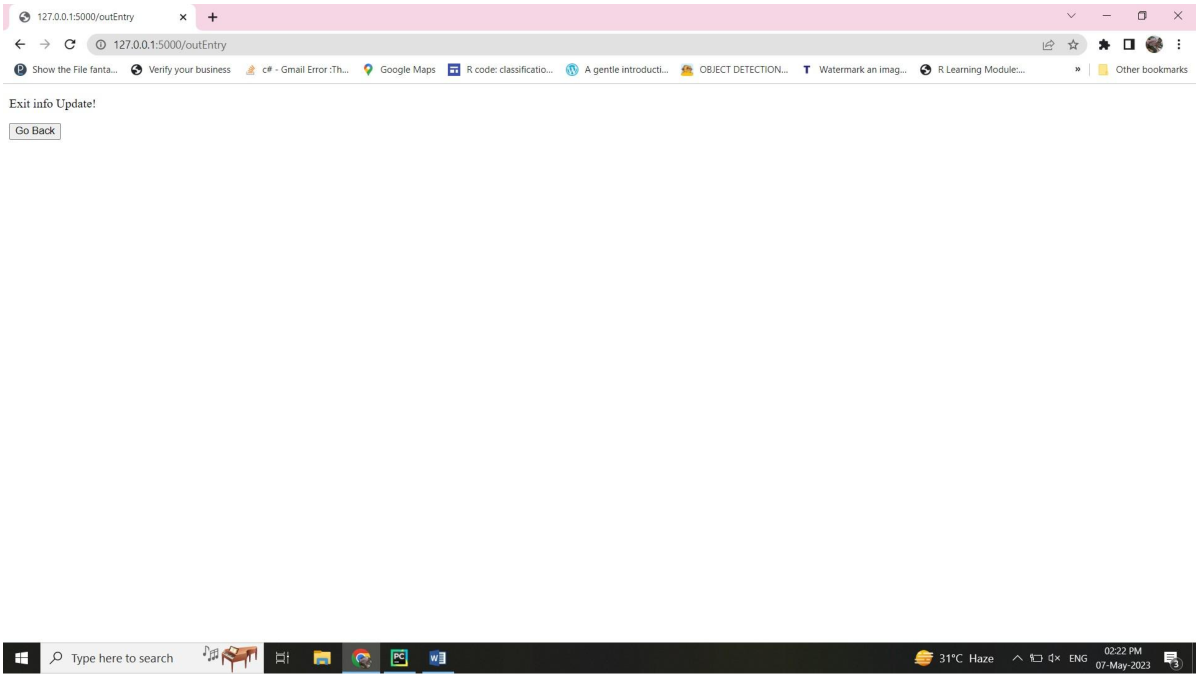




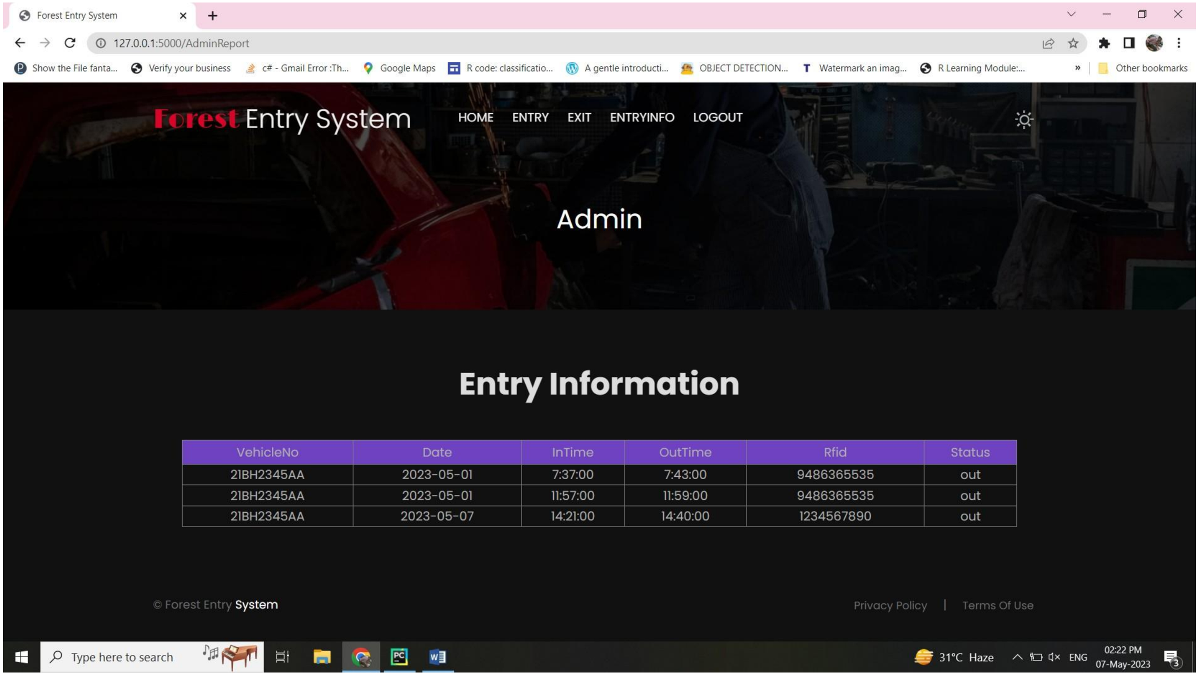
VEHICLE EXIT VERIFICATION



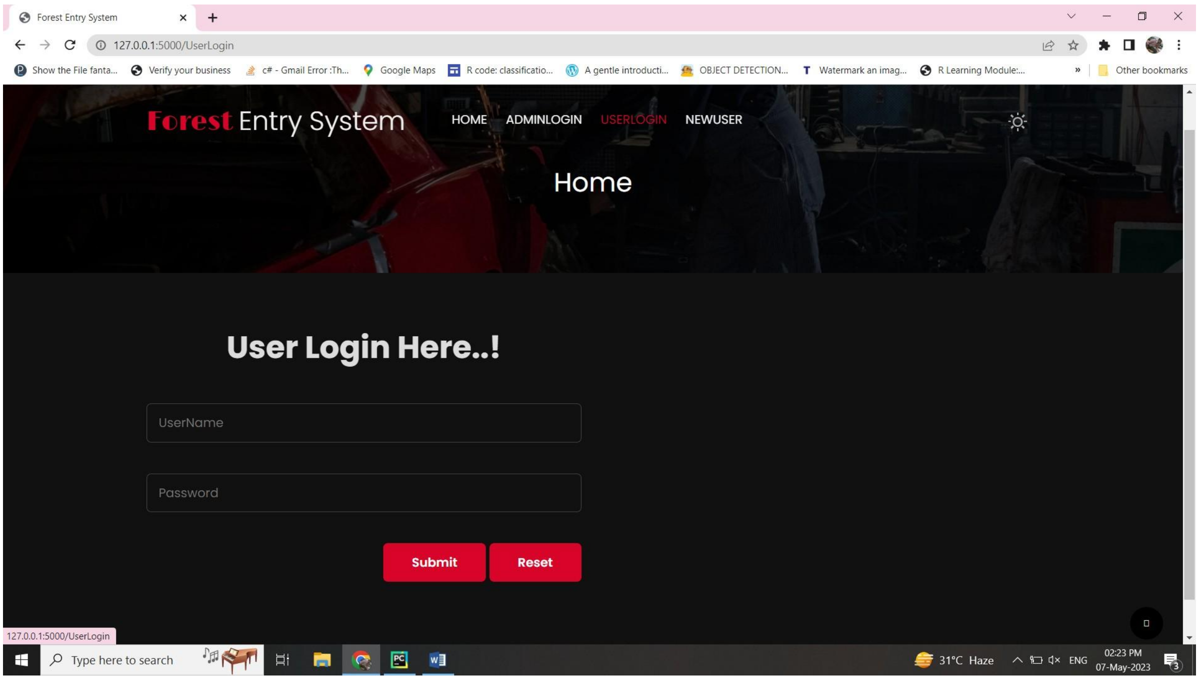

Forest Entry System
127.0.0.1:5000/OutEntry
Forest Entry System
HOME
ENTRY
EXIT
ENTRYINFO
LOGOUT
Exit Verification
1234567890
Submit
Reset
Entry Information

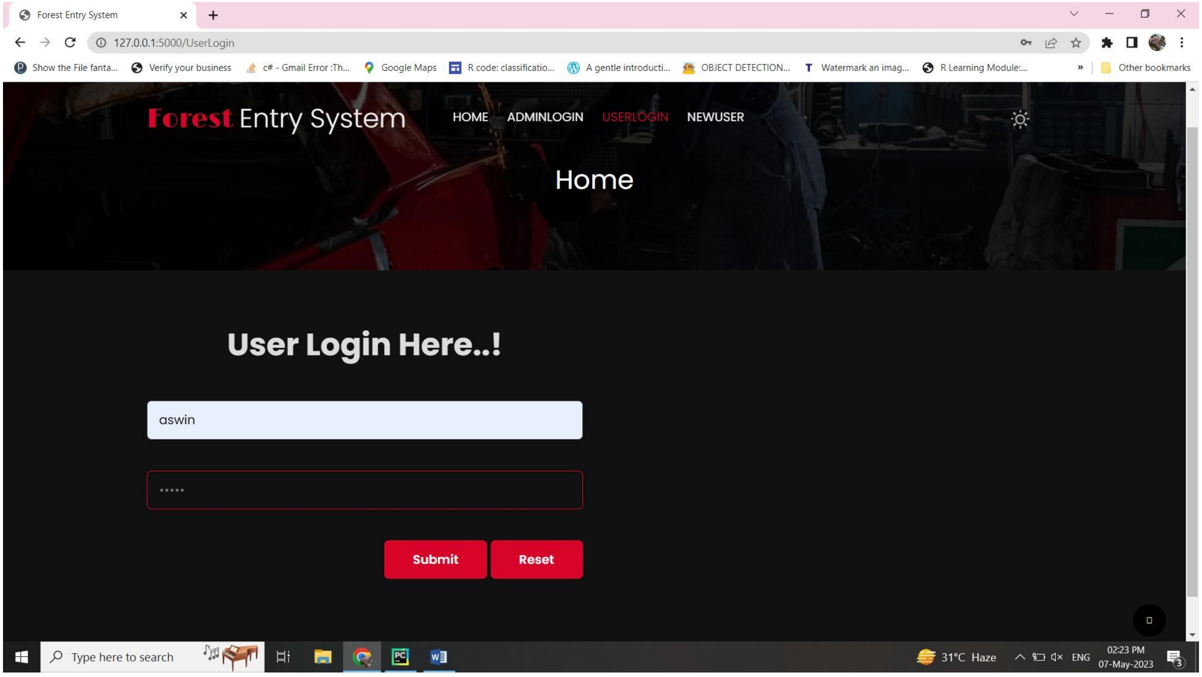

127.0.0.1:5000/outEntry
Exit info Update!
Go Back

ENTRY INFORMATION

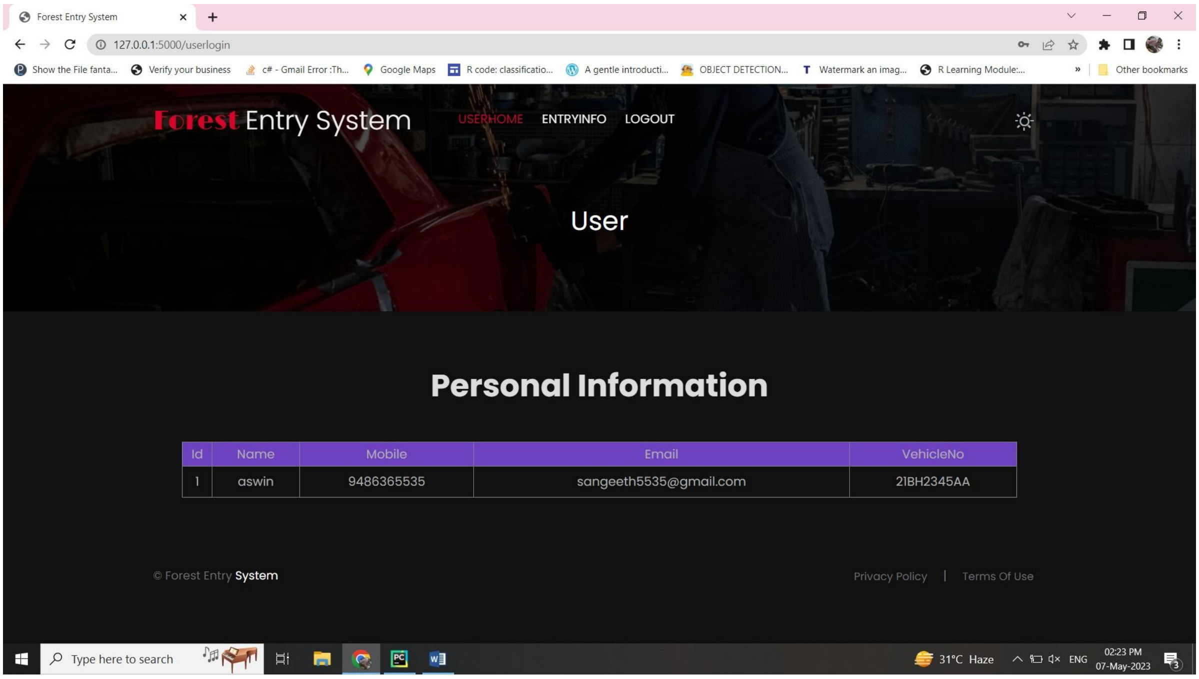


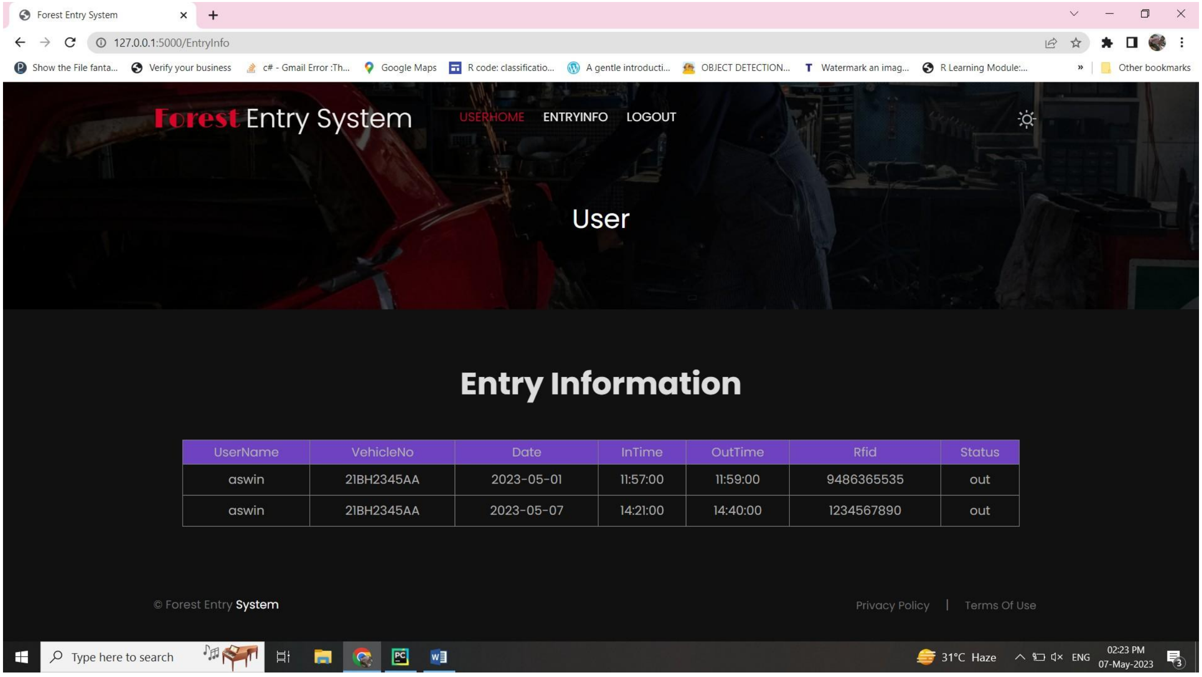
USER LOGIN





USER INFORMATION




Forest Entry System
127.0.0.1:5000/EntryInfo
Forest Entry System
USERHOME
ENTRYINFO
LOGOUT
User
Entry Information
UserName | VehicleNo | Date | InTime | OutTime | Rfid | Status
aswin | 21BH2345AA | 2023-05-01 | 11:57:00 | 11:59:00 | 9486365535 | out
aswin | 21BH2345AA | 2023-05-07 | 14:21:00 | 14:40:00 | 1234567890 | out
© Forest Entry System
Privacy Policy
Terms Of Use

REFERENCES

1. Chen, CL Philip, and Bingshu Wang. "Random-positioned license plate recognition using hybrid broad learning systems and convolutional networks." IEEE Transactions on Intelligent Transportation Systems 23.1 (2022): 444-456.

2. Chen, Joy Iong Zong, and Joy Iong Zong. "Automatic vehicle license plate detection using K-means clustering algorithm and CNN." Journal of Electrical Engineering and Automation 3.1 (2021): 15-23.

3. Shashirangana, Jithmi, et al. "Automated license plate recognition: a survey on methods and techniques." IEEE Access 9 (2020): 11203-11225.

4. Weihong, Wang, and Tu Jiaoyang. "Research on license plate recognition algorithms based on deep learning in a complex environment." IEEE Access 8 (2020): 91661-91675.

5. Zou, Yongjie, et al. "A robust license plate recognition model based on bi-lstm." IEEE Access 8 (2020): 211630-211641.

6. Henry, Chris, Sung Yoon Ahn, and Sang-Woong Lee. "Multinational license plate recognition using generalized character sequence detection." IEEE Access 8 (2020): 35185-35199.

7. Zhang, Linjiang, et al. "A robust attentional framework for license plate recognition in the wild." IEEE Transactions on Intelligent Transportation Systems 22.11 (2020): 6967-6976.

8. Castro‐Zunti, Riel D., Juan Yépez, and Seok‐Bum Ko. "License plate segmentation and recognition system using deep learning and OpenVINO." IET Intelligent Transport Systems 14.2 (2020): 119-126.

9. Tourani, Ali, et al. "A robust deep learning approach for automatic Iranian vehicle license plate detection and recognition for surveillance systems." IEEE Access 8 (2020): 201317-201330.

10. Selmi, Zied, et al. "DELP-DAR system for licens plate detection and recognition." Pattern Recognition Letters 129 (2020): 213-223.