engineering proceedings

MDPI

Proceeding Paper

Electronic Booking and Payment Platforms for Inter-Campus E-Bikes †

Sinovuyo Dunga, Oluwafemi Oni and Omowunmi Mary Longe *

Department of Electrical & Electronic Engineering Science, University of Johannesburg, Johannesburg 2006, South Africa; 219002664@student.uj.ac.za (S.D.); 223248712@student.uj.ac.za (O.O.)

* Correspondence: omowunmil@uj.ac.za

† Presented at the International Conference on Electronics, Engineering Physics and Earth Science (EEPES'23), Kavala, Greece, 21–23 June 2023.

Abstract: In South Africa, traditional transportation has become more expensive and not easily accessible to everyone. The current use of internal combustion engine (ICE) vehicles in the transport sector has been contributing about 14% to the country's greenhouse gas emissions. In booking any e-hauling service, most of the current transportation requires pre-booking or standing in long queues to make a booking. These transactions require usually physical cash or insertion of card details online to process a booking payment. The outcome is prone to hackers and theft. Therefore, this paper focuses on the design and building of a safe electronic booking and payment application software for inter-campus e-bikes through a Quick Response (QR) code and Near-Field Communication (NFC). The application also encourages the use and integration of clean energy solutions in the transport sector and the saving of waiting time for inter-campus ICE buses as students would rather use e-bikes for their inter-campus mobility. Also, they can transact safely, affordably, and securely for their mobility costs using the developed application.

Keywords: quick response; near-field communication; internal combustion engine; electric bike; booking app

check for updates

Citation: Dunga, S.; Oni, O.; Longe, O.M. Electronic Booking and Payment Platforms for Inter-Campus E-Bikes. *Eng. Proc.* **2023**, *41*, 12. https://doi.org/10.3390/engproc2023041012

Academic Editor: Grigor Mihaylov

Published: 14 July 2023

Copyright: © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https://creativecommons.org/licenses/by/4.0/).

1. Introduction

In South Africa, there has been an increase in demand for easy, accessible, and cheap transportation [1]. The traditional means of transportation through fossil fuels releases pollutants, which have negative impacts on the environment and climate change. E-hailing services have been introduced and adopted by different communities and have proven to be thriving. These services are convenient; however, they do not solve the traffic congestion we have and do not make use of renewable energy sources and, thus, are mostly affected during rotational loadshedding.

Existing mobile applications for e-hailing services including rental and online payments allow transaction payments to only be made when a user is online and through the insertion of card details online. However, the recent malicious acts of hackers have shown that this means of transaction has proven to be unsafe, exposing the user to scams. In addition, it puts the user at risk for fraud, and credit or debit card theft, and customers' details can be used for malicious activities without their consent. Third-party access to this vital information can lead to the data being sold or used for fraudulent activities. However, with the evolution of smart cities, hard cash has become the least used form of payment for services. Transactions and tasks are currently being digitalized [2].

In Asian countries, the instant use of Quick Response (QR) payments for purchasing items from vendors has been recorded. So, vendors do not incur additional costs to pay to card companies. This is convenient for users as they are not exposed to security breaches associated with card payments and, moreover, encourages the use of smart devices [3]. The

use of QR and Near-Field Communication (NFC) for payments, as shown in the literature, encourages the use of smart technologies that are aligned with the Fourth Industrial Revolution (4IR). The adoption of these technologies acts as a gateway for new research in renewable energy and the interlink of technologies that are advanced and easy to use. In South Africa, the adoption of these technologies is an advantage as we are at the peak of going green by 2030 according to the World Bank [4]. It provides opportunities for employment and the reduction in traffic congestion.

Quick Response (QR) is a two-dimensional barcode that stores data. It allows for contactless payment where one scans the QR code from a mobile application. When the QR code is scanned, the horizontal and vertical patterns are decoded by the software of the smart devices into a string of characters that allows for confirmation of payment [5]. Near-Field Communication (NFC) is a wireless communication technology that allows data transfers over a distance of 4 cm based on the ISO/IEC 18092 standards [6]. It also allows for contactless payments and bookings. These technologies are currently being deployed in most of the developed countries. However, the usage in South Africa for e-hailing services is limited. And the focus on the adoption into e-bike renting has not been recorded in the literature.

Therefore, the aim of this study is to carry out a detailed design of a booking and payment mobile application using a Quick Response code and Near-Field Communication for inter-campus electric bikes to ensure the availability of an inexpensive and convenient renewable mode of transportation. This was performed to reduce the stress experienced and time that students spend while transporting from one university campus to another. Electric bikes are not only beneficial environmentally, but also economically. While automating the process of renting and payment for the use of e-bikes, as well as avoiding traffic congestion during peak hours, a QR and NFC booking and payment application for electric bikes was designed and implemented. The system's front-end (client side) and backend were designed using open-source technology. Object-orientated programming was used as one of the programming languages for development. This allows for fast computation and reliability of the system. This application would also save each student an average of 10 h weekly. This is useful time that the students can spend on productive activities such as reading and assignments.

2. Theoretical Methodology

2.1. QR and NFC Code Payments

This is a specific matrix barcode that is two-dimensional, and the data are stored and read vertically or horizontally. The encoded information can be readable through smartphones that have a high-resolution camera or any barcode readers. The system consists of black modules that are arranged in a square pattern on a white background. The information encoded can be either in text, URLs, alphanumerical numbers, or any form of data [7]. It can be readable from any direction if the GUI is clear.

Information is submitted by users to the online merchant and is minimized by providing minimal information that is encrypted. Unique QR codes can be generated for each new transaction without having to carry card details. The QR code is generated with all the necessary details such as the customer's name, CVV, expiry date, and card number. The QR code is updated with the specific amount of the rental of the e-bike that is submitted to the payment system. The user's details are verified by the payment gateway provider [8]. This type of payment is secure, contactless, offers a better customer experience, and has no additional fees to card companies. The technology is used for many applications such as product tracking, identification of products, time tracking, as well as making payments. This type of technology allows for instant payments and has high security because of the encryption that makes it reliable.

Table 1 displays the QR code specifications that need to be considered when designing the application. The Reed Solomon error detection and correction algorithm (RS) is used to recover from transmission errors. It has four error correction levels, namely, low, medium,

quality, and high. It enables the QR code information to be readable even if some parts are damaged. The Kanji and kana represent the character modes, number mode, alphanumeric mode, and the 8-bit byte mode [8]. This method of payment is preferred over the traditional use of physical cards and cash because of the encryption of the user's information and the fact that no details are stored in the file.

Table 1. The QR code specification.

Item	Specifications	
Error-correcting codes Row Name Here	RS code BCH code	Data Formation and Version Information
Characters	Number Alphanumeric 8-bit byte Kanji	10 bits coding per 3 number digits 11 bits coding per 2 characters 8 bits coding 13 bits coding per 2 characters
Version	1 2 40	21 × 21 modules 25 × 25 modules 177 × 177 modules
Error-correcting level	L M Q H	About 7% About 15% About 25% About 30%
Finder pattern	1:1:3:1:1	3 concentric squares 7 × 7, 5 × 5, 3 × 3 modules
Alignment pattern	1:1:1:1:1	Higher version 2 5 × 5, 3 × 3, 1 × 1 modules

On the other hand, NFC technology uses electromagnetic radiation to transmit information between two NFC-enabled devices. It is contactless and allows for an exchange of data over a short range. Transactions are carried out with little energy and when the payment device comes into contact with the NFC reader, both are activated and the exchange of encrypted information causes the payment.

Some of the risks of devices becoming exposed during payment are reduced by the ability of NFC to connect to only one mobile device. This reduces the risks of an individual paying someone else's bill. When an individual loses their device, the transactions can be blocked from the Primary Account Number (PAN) by disabling the information from the device. The two-step authentication technology prevents fraudulent activities that could occur. This is conducted through encryption and decryption on the authorized device. The included methods enhance security in the use of the technology. Figure 1 presents the process of accepting transactions from the credit or debit card using a QR code and NFC technology.

The NFC-enabled device has a secure element architecture that protects the execution environment for applications where users can store their personal and financial information. It has a tamper-resistant microprocessor that provides secure authentication and memory for storage payment applications. It contains hardware, software, protocols, and interfaces that enhance security. The memory of the SE is hosted on the baseband processor of the smart device. The baseband processor manages cell phone connectivity and application operation [9]. For Android devices, a Host Card Emulation feature is included instead of the SE for secure storage of account information. Instead of the information being stored in the device itself, it is stored in the cloud. However, some of the drawbacks of the technology include it operating over a short distance of 4 cm and offering low data transfer rates that are either 212 or 424 kbps [10]. The technology is quite expensive and has complex procedures; thus, most users do not make use of it. The communication mode allows the target and initiator devices to have the power supplies and can communicate with one another by alternating the signal for an active mode.

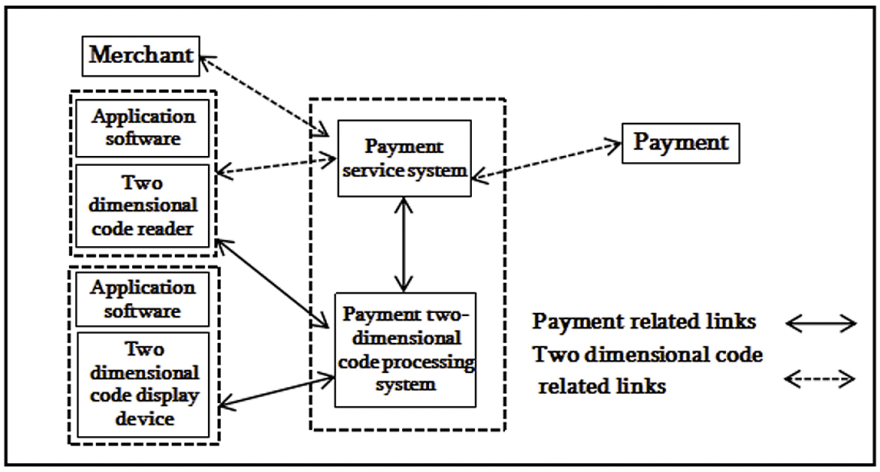


Figure 1. QR code architecture with payment gateway providers [11].

2.2. ASP.Net Framework

This is an open source that is suitable for creating web applications that can be processed on a server and run on any browser. This was created by Microsoft to run an application on Windows. The language used is .Net and the library accessed is from the framework class library and the common language runtime for efficiency of the code. The common language runtime is vital for exceptions, which are errors that occur when running the application. Identification of the CLR helps with debugging errors. The framework considers unwanted resources and uses the garbage collection principle to regularly check and remove them to create a memory. There is also a security feature for verifying and validating the developed applications. Users will need to be granted access to security mechanisms [12]. The ASP.Net framework uses the C# programming language that is object-orientated programming. It is fast, reliable, and simple to use. .NET is the development platform for running and executing applications. The code is compiled and translated to object-orientated code and it takes little time for execution. The compilation process validates the consistency of the code, which makes it easy to fix the error. The framework is also compatible with Xamarin for mobile application designs.

2.3. Microservices: Architectural Design

Microservices address issues related to the architecture and the patterns involved when working with microservices. It focuses on classifying the large applications and each microservice is responsible for addressing the aspect and functions of the application. When compared to the RESTful services, the microservices APIs are faster and defined by the protocol buffer that exchanges data [13]. The ASP.NET project can be initialized on the microservices by using the dotnet command line such as the dotnet new webapp –o BasicGrpcService. This is a new function that acts as an inter-process communication in the microservices between the data centers with pluggable support for load balancing and authentication. Some of the benefits of using microservices include it being small and taking less time for development. The environment is faster, easy to maintain, and has fewer errors. If one service is not functional, it does not affect the rest of the system. The microservices cover the following elements in the database:

- User account management;
- Booking component for the e-bike;
- Payment components are the QR code and NFC method;
- Data storage for sensitive information for authentication such as hashed passwords, e-mail addresses, contact details.

Other interfaces needed for the implementation of this study include (a) Microsoft SQL, which is responsible for the relational database. (b) Visual Studio is a web development

tool that uses HyperText Markup Language (HTML). (c) Cascading Style Sheet (CSS) and JavaScript to design and develop a web-based application. NodeJS was also used for the backend interface with Xamarin. (d) The ADO.Net technology is used for interacting with the applications using the database Microsoft SQL. The framework shall be hosted on Microsoft Azure. (e) Xamarin is a native language used to develop mobile applications for Android, iOS, and Windows devices with .NET. (f) Stripe, a payment gateway provider, is used to access the visual studio using already installed packages and it is connected to a publishable key [11].

3. Design Flowchart

The detailed design is designed on ASP.NET. The Data Flow Diagram (DFD) shown in Figures 2–10 presents the process of booking an e-bike ride through the mobile application. The oval shape represents the entity or user. The boxes represent the process that shows the number of processes and its description. Figure 2 displays the homepage of the mobile application for the three actors: admin, customers, and the visitor. The visitors are those who are visiting this app homepage for the first time, while the customers are the returning clients who have already signed up on the booking page. Figure 3 shows the user booking window before confirming booking while Figure 4 shows the administrator database page.

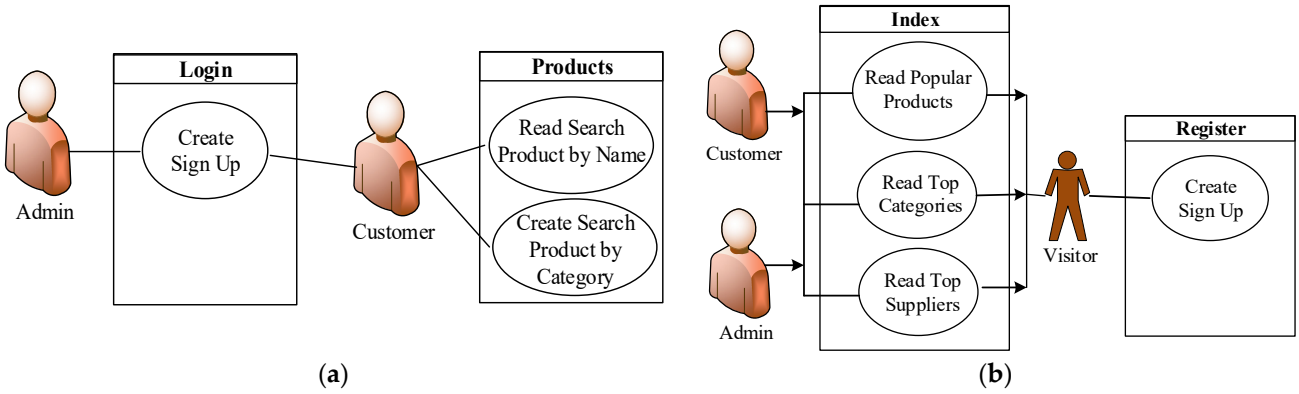


Figure 2. Design of the homepage for the application (**a**) Setup for customer and admin; (**b**) setup for the admin, an existing user as customer and for a new client visiting the page.

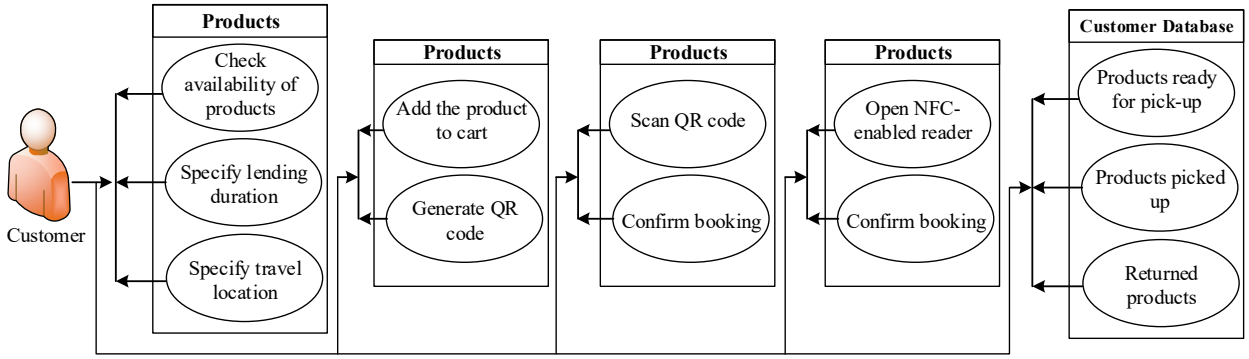


Figure 3. The user booking window, customer cart page, booking confirmation using QR or NFC, and database page.

During the DFD Level 0 phase, the layout of the architectural design for the mobile application is presented. The context diagram describes how the project interacts with the environment. The external entities and data flowing in and out of the system are clearly outlaid. Figures 5 and 6 show the NFC/QR booking and Payment architectural structure.

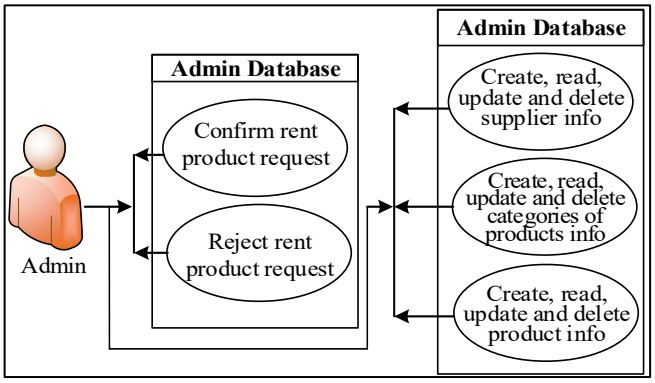


Figure 4. Administrator database page.

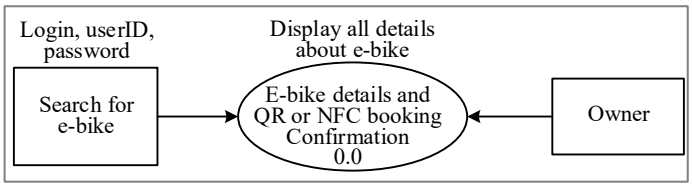


Figure 5. Interface between user and owner (QR/NFC).

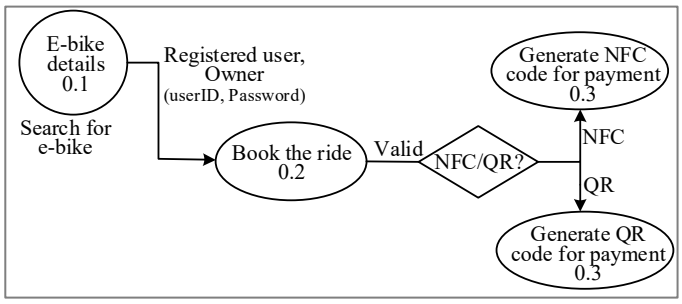


Figure 6. E-bike booking using QR or NFC method.

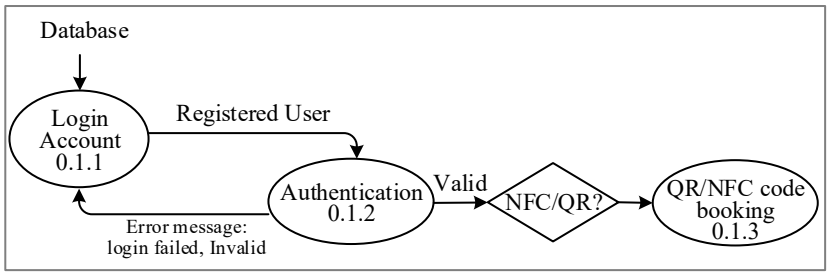


Figure 7. DFD for registered user (QR or NFC booking).

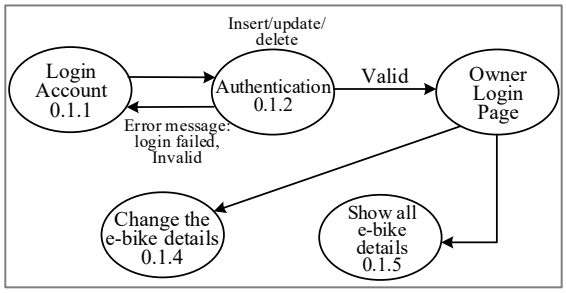


Figure 8. Administrator dataflow design.

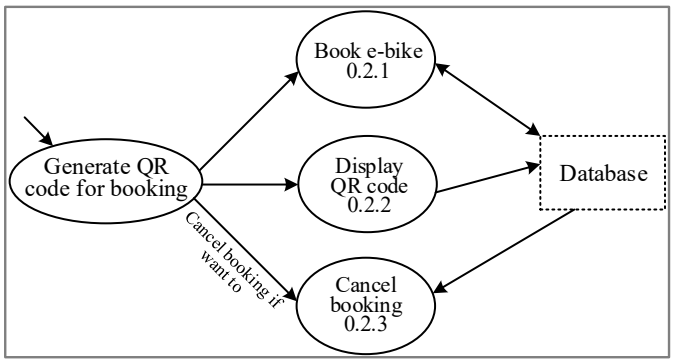


Figure 9. The booking procedure for authenticated users.

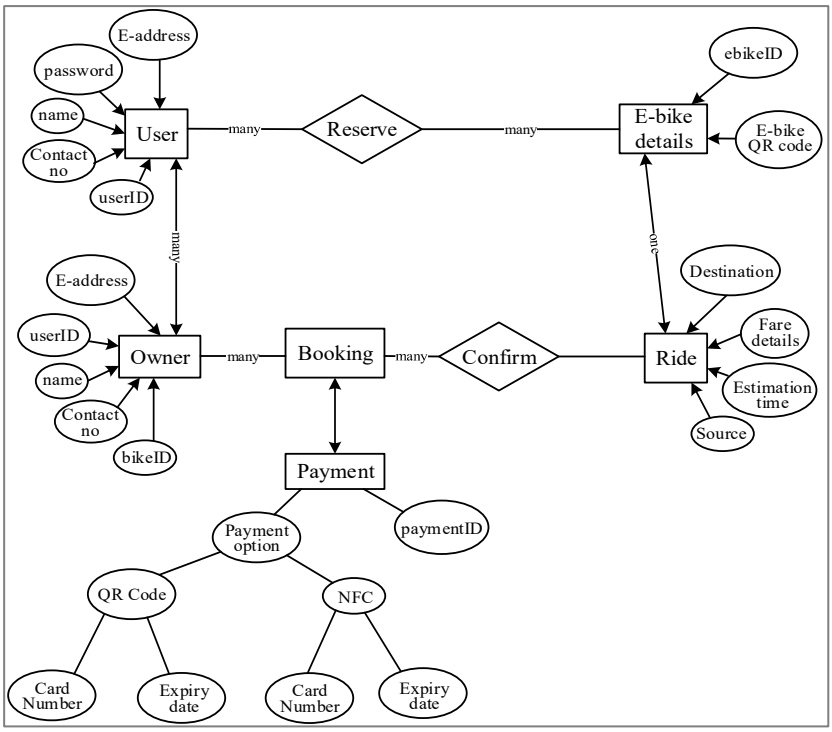


Figure 10. Design of database table.

During the DFD Level, the data flow diagram for this stage is used to register or decline new users. On the admin database, the availability of e-bikes can be confirmed, as well as whether an existing user can gain access. The DFD is shown in Figure 7. This diagram shows the DFD for a Registered User using a QR/NFC code. Figure 8 shows how the data flow diagram is designed for the administrator side.

During the Data Flow Diagram Level 2, Figure 9 shows the data diagram design for the application. The relationship diagram, Figure 10, shows the entity relationship diagram that represents the visual presentation that facilitates the database by allowing the specifications of the mobile application schema to represent the overall structure of the data. This diagram also shows the class diagram that represents the structure of the system. It displays the systems classes, attributes, and operations within the environment as well as the relationship among the system.

4. Testing and Results

There has been an increase in demand for contactless and instant payment applications that use new technology such as Quick Response and Near-Field Communication. These technologies minimize costs related to the technology use and are highly secured through

encryption. A user can make the payment offline by scanning the QR code that takes them to a third-party payment gateway to process the payment. The method promotes hygiene, especially in pandemic situations. It is also safer; administrators are not required to keep records of payment details and it increases operational efficiency.

Therefore, in the simulation of different scenarios, different contingencies are put in place to ascertain the quality and assurance of this design and to present an overview of the design specifications and coding. The overall purpose is to verify and validate the various microservices to ensure that the requirements are met.

During the validation stages, different rules and regulation are put in place:

- Records cannot be taken until all the necessary entities are performed.
- Only the owner can edit the e-bike and user details, register new users, and/or delete existing members.
- For security reasons, the user's e-mail address and contact details are required for cases when login details are forgotten and need to be retrieved.

The experiments are divided into two sections: functionality, which is used to evaluate whether the system works as expected, and performance, which evaluates how well the system works with the design. Experiments are conducted for the following subsystems: Index page, Home page, Register page, Login page, Products, Cart, Checkout, and Firestore Database.

The number of tables utilized and their architecture layout are determined by the logical relationship in the data supplied. The database model outlines the numerous tables that are utilized as well as the data that are kept in the tables. Formats of the fields and their types of data structure as well as the user information are also taken into consideration, as shown in Tables 2 and 3. The table represents the database and the data types that are used for the design of the mobile application.

Table 2. Tables in the database.

Attributes	Data Type
First name, Last name, UserID. Password	Varchar
Contact No	Numeric
Hint Question, Hint Answer, E-mail ID	Varchar
UserID, User First name, User Last name, User e-mail ID	Varchar
User contact no	Numeric

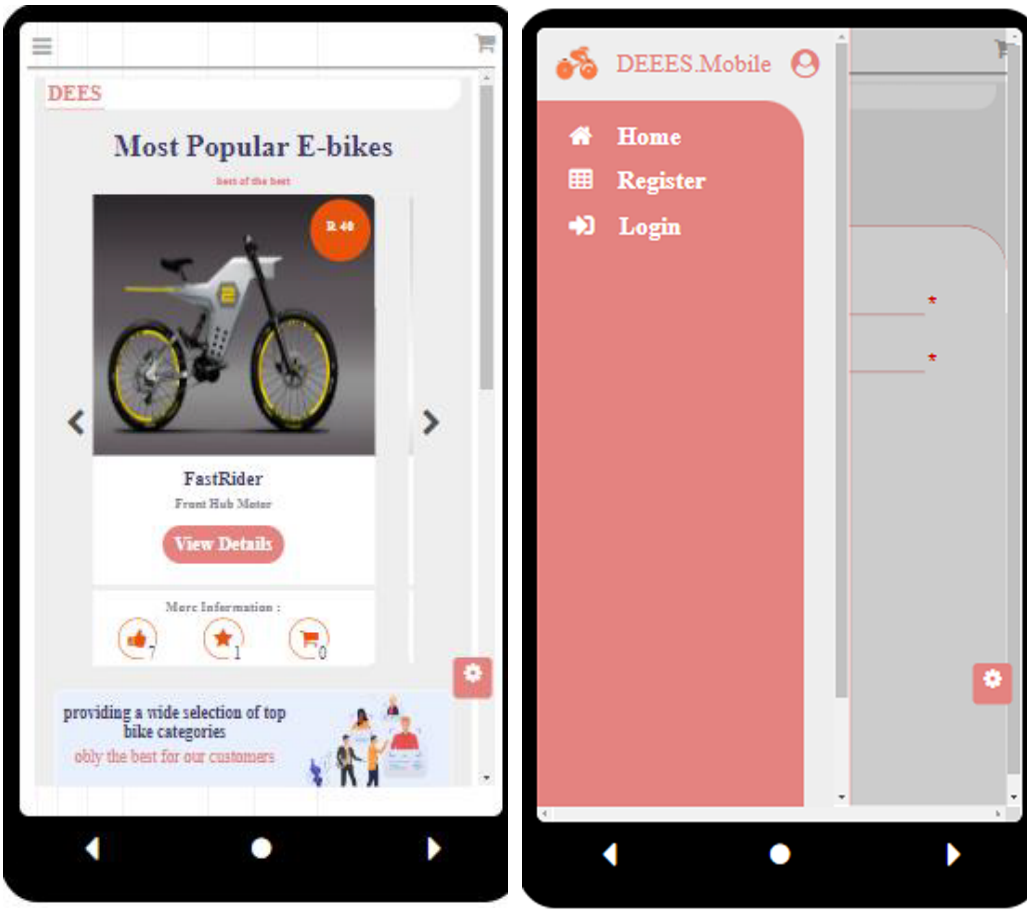
Table 3. Booking information.

Attributes	Data Type
EbikeID, Ebike QR code	Varchar
Number of debit card	Numeric
Type, Name, UserID, ExpDate, QR code	Varchar

This is where the user interacts with the application before registration and rental of the electric bike. HyperText Markup Language (HTML) is used to structure the web pages and the content for the application. The .html extension contains information about the landing page with information about the products, categories, and the most used e-bikes.

The application displays the front page containing information about the different electric bikes for use with the respective information. The user can scroll on the sides to view the different e-bikes. Figure 11a displays the client-side page with the different electric bike options for rental. The application displays the home page linked with the index page that contains the layout information about the electric bikes that can be used. The user can scroll on the sides to view the different e-bikes. A front-end page is provided where the visitor can insert their details to be stored in the database. The application registers a visitor successfully and stores their details on the firestore database (Figure 11b).

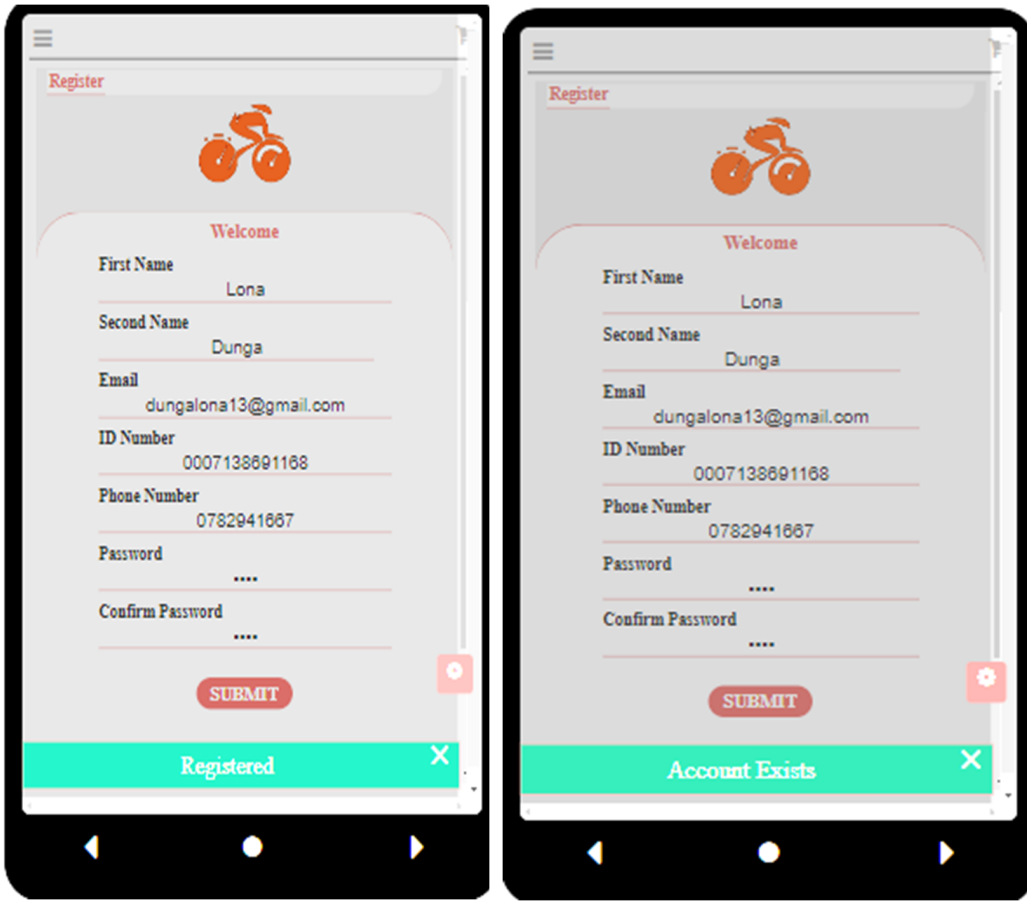
Figure 11b represents the mobile application dashboard that the visitor can see. This page contains the home page to view the different electric bikes and the registration page with the registration form to become a new customer. The purpose of the login page is to allow the user to gain access to the application by entering their e-mail address and password, as shown in Figure 12a. The application is expected to allow the registered customer to log in. The database is expected to store the account details of the user. In a scenario where an already registered user refills the register form, a message that says that the account exists will pop-up (as shown in Figure 12) because the database holds the details (Figure 13).



(**a**) (**b**)

Figure 11. The home page (**a**) and the login page of the application (**b**).

The purpose of the product page is to display the different electric bikes. The search button can be used to search for an electric bike by either name, price, or category. The programming of this study allows for the products to be selected and be added to the cart. The Quick Response code connected to the different products is also created in javascript. The images of the bikes are stored in a string and a QR code can be generated with the different products for payment. The purpose of the cart page is to allow the user to see the products that have been added to the rental cart. It contains the details of the electric bikes and the price before the customer pays.



(**a**) (**b**)

Figure 12. The user account showing (**a**) registered after input of customer details, (**b**) already exists in the database.

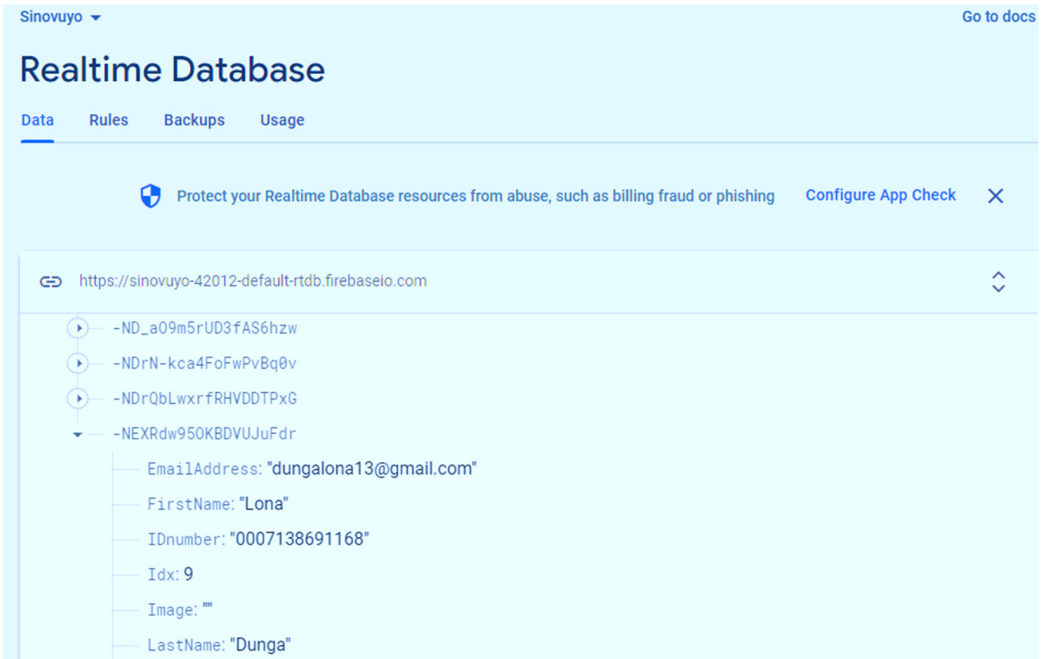


Figure 13. Real-time database.

A products page file with an extension file of .html is created in visual studio code. The customer adds the electric bikes to the cart. The application asks the user to insert the user time, start location, and the destination location. After the user has inserted the details, a list of the added products is displayed on the screen with the estimated price before the customer pays for the rental. The database is created on firestore and one of the limitations of using the database includes the 1 MB limit per document. The user details are saved in the database (details in Appendix A). This work contributes toward the role that the electronic booking and payment system would have in attaining the Just Energy Transition goal of the government of South Africa in the transport sector [14–16]. Furthermore, it also saves an average of 10 h weekly per student spent in the queue for campus buses. The students can use this saved time to perform other productive activities that can enhance their academic performance and social development. Other functionalities of the app are presented in Appendix B.

5. Conclusions

This study made use of HTML, CSS, Javascript, NodeJS and ASP.Net applications to design a seamless coordination of an e-bike rental system. An application software was designed to register visitors, encrypt their details, and make them become users. The users can make payments through a QR code and NFC module via the stripe third payment merchant. The system was able to provide feedback to the administrator on the use of the e-bike. Users can share their experiences via the feedback section and the firestore database is able to show statistical data for use of the e-bikes.

In conclusion, the implementation of the QR and NFC booking and payment platform for e-bikes is a significant step toward improving the overall experience of e-bike riders. This solution simplifies the booking process and enables riders to make payments seamlessly without the need for cash or credit cards. The QR code and NFC technology offer a convenient, fast, and secure way of making transactions, making it an ideal solution for the current digital era in transportation. Moreover, this technology will encourage more people to use e-bikes; reduce traffic congestion on four-wheel trains; make students arrive on time for their lectures, tests, and exams; and promote sustainable transportation. Just as it can offer time savings for the students, it can do the same for others when deployed in a city, hence offering time savings that can be used for other productive and economic activities. Overall, the QR and NFC booking and payment platform is a game-changer for the e-bike industry, and it is expected to revolutionize the way we move around cities.

Author Contributions: Conceptualization, S.D. and O.M.L.; methodology, S.D., O.O. and O.M.L.; software, S.D. and O.M.L.; validation, S.D., O.O. and O.M.L.; formal analysis, S.D., O.O. and O.M.L.; investigation, S.D. and O.M.L.; resources, S.D. and O.M.L.; data curation, S.D.; writing—original draft preparation, O.O. and O.M.L.; writing—review and editing, O.O. and O.M.L.; supervision, O.M.L. All authors have read and agreed to the published version of the manuscript.

Funding: This paper publication is made possible with funding received from the National Research Foundation, South Africa with grant reference SRUG2205025715.

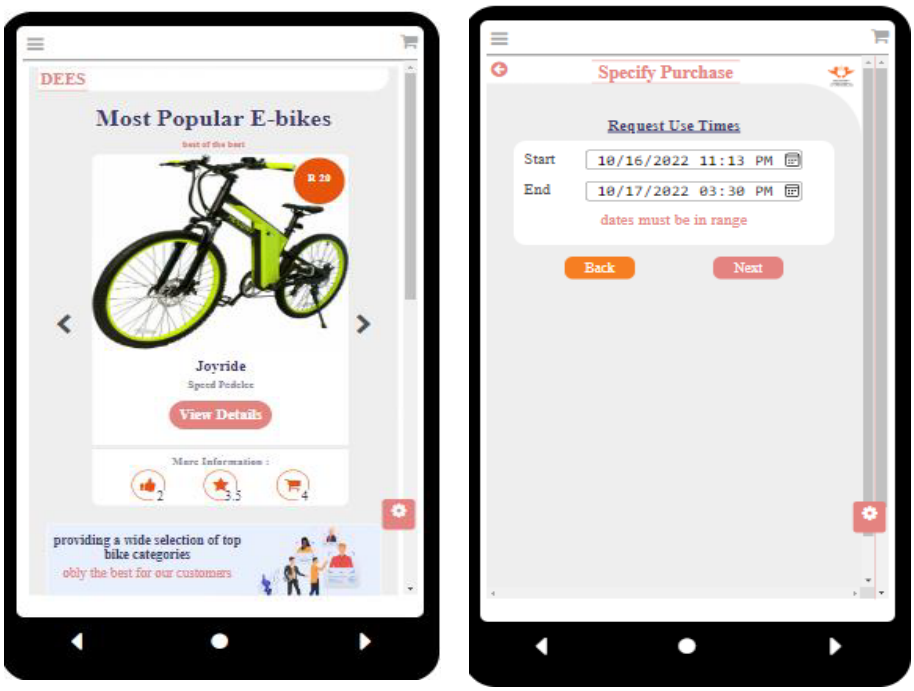
Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Conflicts of Interest: The authors declare no conflict of interest.

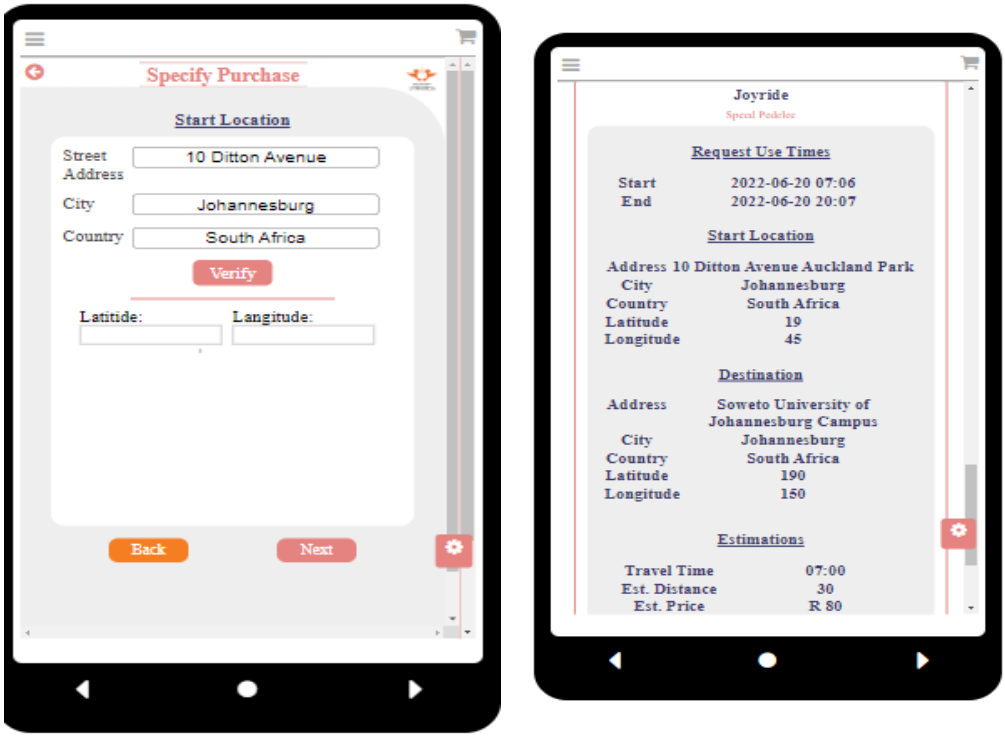
Appendix A



(**a**) (**b**)

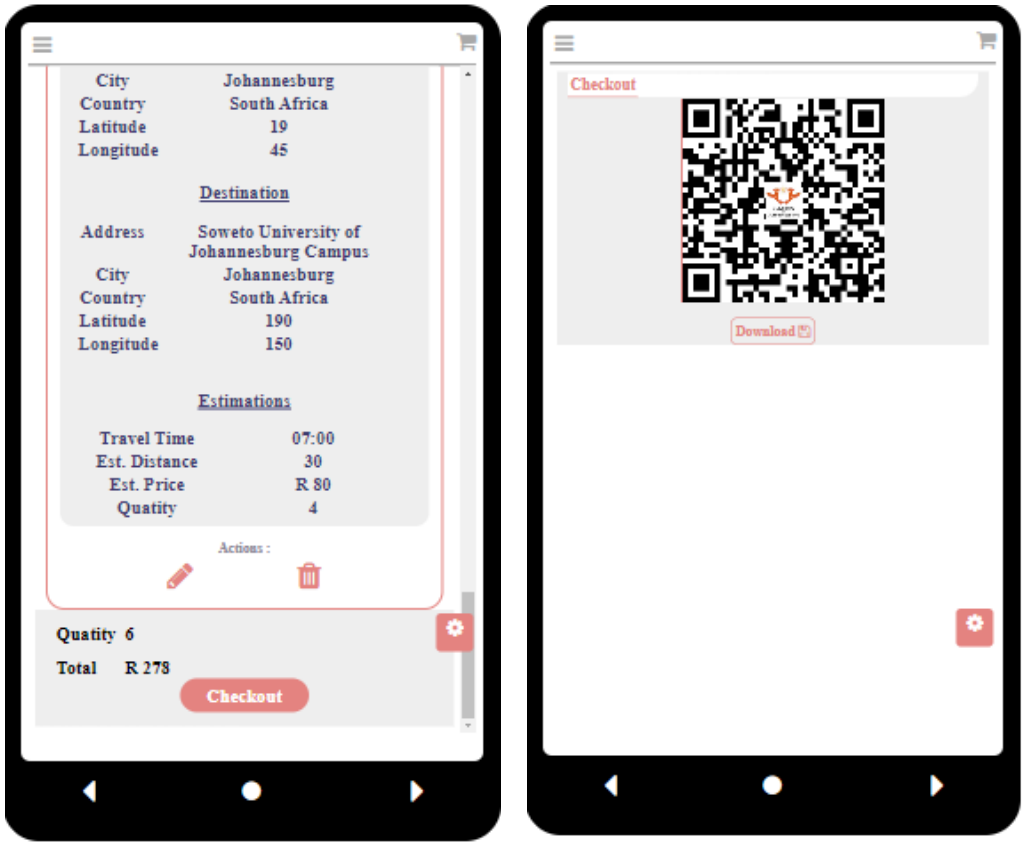
Figure A1. (**a**) The shop page with the different electric bikes, (**b**) select date and time for e-bike rental.

Appendix B



(**a**) (**b**)

Figure A2. (**a**) select the e-bike start location; (**b**) Joyride electric bike receipt.



(**a**) (**b**)

Figure A3. (**a**) the checkout product details for the joyride bike rental; (**b**) the QR code used to check out electric bikes for use.

References

1. Vilakazi, A.M.; Govender, K.K. Exploring public bus service quality in South Africa: A structural equation modelling approach. *J. Transp. Supply Chain Manag.* **2014**, *8*, 1–10. [CrossRef]
2. Gan, Y.; Fan, H.; Jiao, W.; Sun, M. Exploring the influence of E-hailing applications on the taxi industry—From the perspective of the drivers. *ISPRS Int. J. Geo-Inf.* **2021**, *10*, 77. [CrossRef]
3. Hingmire, S.; Mahalle, T.; Bhandekar, S.; Malamkar, N.; Kumar, V.; Zade, P. QR-Code Based Bus Ticketing System with Real Time Tracking. *Int. J. Sci. Res. Comput. Sci. Eng. Inf. Technol.* **2019**, *5*, 636–640.
4. Sebestyén, V.; Domokos, E.; Abonyi, J. Focal points for sustainable development strategies—Text mining-based comparative analysis of voluntary national reviews. *J. Environ. Manag.* **2020**, *263*, 110414. [CrossRef] [PubMed]
5. Narayanan, A.S. QR codes and security solutions. *Int. J. Comput. Sci. Telecommun.* **2012**, *3*, 69–72.
6. Singh, N.K. Near-field communication (NFC): An Alternative to RFID in Libraries. *Inf. Technol. Libr.* **2020**, *39*. [CrossRef]
7. Pourghomi, P.; Ghinea, G. NFC payment applications through cloud computing. In Proceedings of the 2012 International Conference for Internet Technology and Secured Transactions, London, UK, 10–12 December 2012; pp. 772–777.
8. Price, M.J. *C# 8.0 and NET Core 3.0–Modern Cross-Platform Development: Build Applications with C#, .NET Core, Entity Framework Core, ASP.NET Core, and ML.NET Using Visual Studio Code*; Packt Publishing Ltd.: Birmingham, UK, 2019.
9. Martin, R.C. *Clean Code: A Handbook of Agile Software Craftsmanship*; Pearson Education: London, UK, 2009.
10. Jumić, J.; Vuković, M. Analysis of credit card attacks using the NFC technology. In Proceedings of the 2017 40th International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO), Opatija, Croatia, 22–26 May 2017; pp. 1251–1255.
11. Baidong, H.; Yukun, Z. Research on Quickpass Payment Terminal Application System Based on dynamic QR Code. *J. Phys. Conf. Ser.* **2019**, *1164*, 32059. [CrossRef]
12. Yellavula, N. *Hands-On RESTful Web Services with Go: Develop Elegant RESTful APIs with Golang for Microservices and the Cloud*; Packt Publishing Ltd.: Birmingham, UK, 2020.
13. Gabhane, D. Implement Stripe Payment Gateway in ASP.NET MVC. 2018. Available online: https://www.c-sharpcorner.com/blogs/implement-stripe-payment-gateway-in-asp-net-mvc (accessed on 12 April 2023).

14. Semelane, S.; Mkhwebane, E.; Snyman, A.; Letete, T.; Nhlapo, Z.; Ntshalintshali, N.; Purchase-Hatfield, J.; Venter, M.; Passero, A.; Petrosillo, M.; et al. A Just Energy Transition in South Africa. 2018. Available online: https://static1.squarespace.com/static/609a53264723031eccc12e99/t/60ed4d6d4918f36a6d5eb6d3/1626164593677/A+Just+Energy+Transition+in+South+Africa.pdf (accessed on 12 April 2023).
15. Longe, O.M. An expository comparison of electric vehicles and internal combustion engine vehicles in Africa-motivations, challenges and adoption strategies. In Proceedings of the 2022 IEEE PES/IAS PowerAfrica, Kigali, Rwanda, 22–26 August 2022; pp. 1–5.
16. Presicency, T. South Africa's Just Energy Transition Investment Plan (JET IP). 2023. Available online: https://www.thepresidency.gov.za/download/file/fid/2649 (accessed on 12 April 2023).

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.