M-Commerce Offline Payment

Yash Kumar Gupta1 · Girish Jeswani1 · Olan Pinto1

Received: 5 September 2021 / Accepted: 18 November 2021 © The Author(s), under exclusive licence to Springer Nature Singapore Pte Ltd 2021

Abstract
During the ongoing COVID-19 Pandemic we have seen a surge in online transactions. The current wallet payment architecture requires online connectivity during transactions. It is observed that there is a possibility of not getting connected to the Internet due to various reasons. We are working on a solution to enable offline transactions due to any internet failure. We identify three main scenarios where fully offline transaction capability is considered to be beneficial for both merchants and consumers. It includes making purchases in locations without online connectivity, when a reliable connection is not guaranteed, and when it is cheaper to carry out offline transactions due to higher communication/payment processing costs involved in online approvals. Our work aims to address the challenge of providing secure offline transaction capability when there is no online connectivity keeping in mind to not compromise any security features in our M-Commerce offline wallet payment application.

Keywords Security · Software security · M-Commerce · Web sockets · Threats · Document level locking · Asynchronous storage · Information assurance · Hashing · Intrusion attack

Introduction
Mobile payment is the root of financial transactions for most of the services or goods. The majority of the payment system currently in use considers online communication with the network and is very infrastructure dependent, which is very different from wireless mobile Adhoc systems. Payment failure is not an uncommon phenomenon these days, due to insufficient balance or just a lack of internet connectivity. Ensure The customers are unable to establish a connection with the payments platform is also a major reason for online payment failure. However, with the proposed application, end-users will be able to make offline payments using their mobile devices(wallet). The RBI proposed a pilot scheme for small value payments in offline mode with built-in features for safeguarding the interest of the users, liability protection, etc. Under the pilot scheme, banks and payment System Operators will be able to provide offline payment solutions using cards, wallets, or mobile devices for remote or proximity payments. This paper looks into a hybrid online-offline wallet payment system, the online system developed enhances the capability of the current system being used. The offline model will enable users in regions of unstable connectivity, devoid of internet, to successfully onboard in this venture to eliminate hard Cash Transactions.

The upcoming Sect. "Literature Survey" discusses the nature of the present work and compares it to our approach. In Sect. "Methodology" we briefly discuss the methods involved in our proposed approach and how we deal with the security of payments. Section "Results and Discussion" depicts the results obtained and reviews the working of the proposed application.

Literature Survey

The author [1] addresses a challenge in providing secure offline transaction capability when there is no online connectivity on either the mobile or the terminal. It discusses how tokenization can be adopted by the payment industry as a method to prevent Personal Account Number (PAN) and
A novel approach for building secure, scalable, and flexible e-payment systems in offline mode of communication for enhanced security on transactions [3]. They discussed how the current payment systems are directly dependent on the fixed infrastructure of the network (cellular network), which fails to facilitate an optimal level of security for the payment system. The proposed system is based on a simple public key infrastructure for providing security in payment processes. The performance analysis of the proposed model was analyzed and showed that the system is highly robust and secure ensuring anonymity, privacy, non-repudiation offline payment system over wireless Adhoc network. The author [4] analyzed the characteristics of online payment, gives a common online payment system model with emphasis on the discussion of the electronic credit card, electronic cash, and electronic check of three online payment modes, and finally points out the respective application environment. Advances in technology make electronic cash, electronic wallets, online banking, electronic exchange and other methods of payment continue to be mature and put into commercial use.

A method was proposed in [5] based on an e-cash system with multiple denominations that enable e-mERCHANTS to give customers change when in an offline environment. They proposed a system that results in convenient transactions regardless of the transaction amount and reduces the number of e-cash users needed to deposit in advance. They used various security measures namely: unlinkability, verifiability, unforgeability, double spending detection, tamper resistance, and nonrepudiation to provide security while performing the transaction. Open service-oriented architecture for secure mobile transactions was described in [6]. The system was established as a large-scale, secure and complete system comprising several components. It involved mobile network operators, banks, credit card processors, small merchants. The aim was to design a more flexible way to use cash and credit cards securely.

A new prepaid multihop (transferable) cash payment system solution based on hardware technology implementing an electronic wallet (E-wallet) to accommodate digital coins was discussed [7]. The result is a peer-to-peer (P2P) electronic cash transfer equivalent to a physical cash transfer in public use. This paper also incorporated an offline mode of wallet payment but the drawback is that the system needs still a trusted authority which should act as a catalyst to enhance the trust and possible abuse traceability. The authors in [8] outlined the digital wallets’ threats and discussed the requirements of digital wallets that technically addressed the fear of security to customers as usual providers while assisting in the successful implementation of digital wallets. They discussed interlocking between digital wallet user threats and requirements. It also discusses the key security requirements that successful wallet payments need to ensure the users’ safety.

The authors discussed how Micro-payment systems have the potential to provide non-intrusive, high-volume, and low-cost pay-as-you-use services for a wide variety of Web-based applications [9]. It describes a set of extensions, mobile NetPay, optimized for mobile e-commerce. Mobile NetPay provides high performance and security using one-way hashing functions for e-coin encryption. The authors [10] proposed the importance of offline payment solutions, as people tend to behave carelessly in choosing their passwords to avoid the burden of memorizing complex passwords. Such frail password habits are at severe threats to various services available online, especially electronic banking or e-banking. For eradicating the necessity of creating and managing passwords, a variety of solutions are prevalent, the traditional ones being the usage of One-Time-Password (OTP) that refers to a single session/transaction password.

Paper [11] discusses the security of electronic payment systems. This paper focuses on either dominant systems or new attempts and innovations to improve the level of security of electronic payment systems. The survey consists of the Card-Present (CP) transactions and a review of its dominant system i.e. EMV. Several pieces of research at Cambridge university show variants of attacks against this standard, which demonstrates the lack of a secure offline authentication method. This paper proposes the use of smart cards instead of magnetic stripe cards which are not able to participate in the authentication process the evaluation of the EMV migration from RSA cryptosystem to ECC based cryptosystem 3.

Mobile payments offer a great wireless network in e-commerce [12]. This paper discusses the problems such as supporting macro payments, offline transactions, improving the validity of payment in existing e-commerce modes based on mobile payments. This paper puts forward an offline transaction e-commerce system model based on mobile payment which includes the offline POS terminal, a mobile device, and a payment center. In this model, the transaction voucher, a random generated ID on the offline POS terminal is signed by the digital signature technology on the payment center to generate the
payment confirmation. The digital signature verification of payment confirmation, which applies the emerging ID-based cryptography for key agreement and authentication, is the guarantee of validity in offline transactions. The offline transaction model based on mobile payment saves the costs of wiring and also makes the transaction process more convenient. They propose a fully offline payment technique which is a great alternative for e-commerce payments. However, it is limited to customer–merchant transactions and requires additional hardware like the POS terminal. Using mobile wallets instead can provide flexibility for day-to-day transactions as well.

The authors in [13] provide a comprehensive review of secure mobile payment where classification of mobile payment into TPC(third-party payment company)-led mobile payment and Bank-led mobile payment, and based on this, summarize the system structure of mobile payment was performed. The research conducted introduces secure technologies(hardware and software) used in these procedures, discusses and analyzes the security issues that they have been encountered, summarise open issues, and proposes future development directions

This research [14] considers earlier literature and consumer conduct through ground-level overviews and interviews. They likewise investigate the cryptic challenges looked by clients of various age groups towards a complete acknowledgment of digital payments and modes. As a result of these discoveries, an improvement scope has been talked about wherein a few favorite ideas have been made, including, digital biometric authentication, a coalition of various digital payment platforms, implicit redressal, and shared wallets.

Paper [15] highlights the previously existing mobile device features like short message services and bluetooth that have the capability to perform digital money transactions in an offline environment. The methodology revolves around the fact that e-paycards can be used to send or receive digital money with the existing functionality of mobile devices like messaging and Bluetooth to perform financial transactions without delay even when the e-wallet server is not responsive at a given time. The papers findings suggest that e-paycards can be used by merchants and customers and that customers can send these e-paycards to the merchant through any mobile device messaging functionality or through Bluetooth.

This paper [16] introduces a concept of hard wallet for generating trust. Payment that is issued from the hard wallet can be consumed by a second hard wallet, which will further pay to a third hard wallet thereby creating a payment cycle or ecology of digital money for long periods without the use of a telecommunication channel. The hard wallet approach may be personalized and fitted with ownership security capability.

This paper [17], implemented a novel protocol where the proposed offline e-payment scheme satisfies all essential security requirements through simplified computations. In the proposed model, blind signatures ensure the anonymity of the customer sensibly and directly. In doing so, the customer is kept anonymous as long as they are not liable for multiple spending. The implemented protocol takes care of multiple-spending detection, untraceability between the e-coin and its customer, no-swindling, fraudster control, unforgeability of the e-coin, no impersonation, no framing attack.

Paper [18] develops a novel framework which includes features like the low cost technology, easy implementation and operability, usefulness to local people, and scope for creative appropriation. The aim of the paper was to conduct an in-depth field study through interviews with merchants and consumers to evaluate the framework. The developed framework suggested that along with the features like low-cost, easy installation and operability, customers reported that the mobile payment technology, unlike cash transactions, can also afford creative uses such as reflection on past expenses. The research done on this paper reveals developmental assertions towards the need for greater inclusion of local small-scale vendors.

This paper [19] conducted a thorough research and explored multiple proposed models of payment systems through mobile devices, their payment methods, security mechanisms, technologies and comparison mechanisms involved in mobile payment systems. The paper also provides an analysis of the encryption technologies, authentication methods and firewall in mobile payment system.

This paper [20] discusses in detail the different architectures of mobile payment available. Although mobile payment has gained its popularity in many regions due to its convenience, it also faces many threats and security challenges. This paper presents a mobile payment processing model and introduces each type of mobile payment systems. It summarizes the security services desired in mobile payment systems and also the security mechanisms which are currently in place. It further identifies and discusses three security threats, i.e., malware, SSL/TLS vulnerabilities, and data breaches, and four security challenges, i.e., malware detection, multi-factor authentication, data breach prevention, and fraud detection and prevention, in mobile payment systems.

Paper [21] reviews the security issues involved with electronic payments. The paper focuses its research on online payments as well as E-wallets. This research reviews around 131 research papers concerned with electronic payments in between 2010 and 2020. It uses a systematic approach to its literature review. The results obtained show that interest in E-wallet and online payment has grown significantly during 2010–2020 period. The paper proposes that the underlying...
issues with security of online payments should have some protection properties such as authorization, availability, integrity, authentication etc.

This paper [22] compares existing literature regarding mobile payments to enhance understanding of the current state of knowledge. This research paper discusses in detail the peer reviewed surveys available in the Scopus database between the period of 2005 and 2020. About 56 studies published in 44 international journals were used for a systematic literature search in the Scopus database. The paper also conducts a bibliometric analysis with the VOS viewer to identify thematic clusters. Overall the paper provides a decent understanding of mobile payment architectures and based on the results future avenues were also proposed.

After thorough research on existing work, many methods are available for the secure mobile transactions through online. The usage of OTPs, tokenization, EMVs for the secure transactions were discussed. Using this as a base, we came up with a solution considering security at high level for the mobile payments. The main drawback in the literature was the lack of offline payments implementation. The idea of offline payments is brought up in literature but none of them have created an end to end offline mobile payment system. We considered this concept for the implementation of safe and secure online-offline payment system (Table 1).

### Methodology

A novel application that incorporates an M-Commerce wallet payment system in an offline environment. This application was developed with a particular goal that is to enhance and improve the current wallet payment system between a customer and a merchant without compromising any security features (Fig. 1). The application incorporates online as well as offline payment methods. To carry out this, a well designed M-Commerce payment architecture was developed to prevent any security threats (Fig. 2).

In the case of online payments, the first step is a user login page, that has a safe API protocol whereby the end

### Table 1 Comparison with existing methods

| Sr. No. | Paper                                                                 | Comparison                                                                                           |
|--------|----------------------------------------------------------------------|------------------------------------------------------------------------------------------------------|
| 1      | Research on offline transaction model in mobile payment system [11]    | This paper proposes an offline transaction e-commerce system model which includes an offline POS terminal, mobile device and a payment center. The key idea of this model is using the mobile device as a medium to transfer the offline terminals transaction voucher and payment center’s payment confirmation to complete the transaction. This paper offers a fully offline method of payment, that is convenient and cost effective. However, we propose an application that offers wallet based mobile payments which can be conducted irrespective of the availability of internet connection (online as well as offline) |
| 2      | The impact of pandemic on digital payments in India [23]              | This paper tries to highlight how the pandemic has fueled India towards Digital Payments due to fast and easy transactions, added security and transparency. It discusses how the population has moved on to online transaction payment systems such as UPI and Mobile Wallet payments. They discuss various forms of digital payment techniques which provided us with a strong foundation for our paper |
| 3      | Offline transferable E-cash mechanism [5]                           | This paper proposes a system that results in convenient transactions regardless of the transaction amount and reduces the number of e-cash users needed to deposit in advance. However, there is limitations to using e-cash as these transactions are usually complicated to trace and since it is a digital currency, it is prone to forgery as well |
| 4      | Privacy-preserving offline mobile payment protocol based on NFC [2]   | This paper proposes a new privacy-preserving offline mobile payment protocol based on NFC. This scheme is well designed to accomplish all security and privacy requirements. Even though this approach is convenient, not all devices support the use of NFC in the Indian market |
| 5      | Review of digital wallet requirements [8]                           | This paper highlights a novel approach for building a secure, scalable, and flexible e-payment systems in offline mode of communication for enhanced security on transaction. Looking at the existing architectures of mobile payment systems, we took inspiration from several wallet based applications to develop an extensible and secure application of our own |
users credentials are hashed and stored in the database. The login page is essential for user identification and to prevent identity theft. As a developer, we need to keep user data as safe as possible even in the situation of a data breach, hence hashing key user information is crucial.

We need a promise-based HTTP client that returns promises rather than accepting callback. Axios is a popular, promise-based HTTP client that supports an easy-to-use API and can be used in both the browser and Node.js. This ensures handling asynchronous calls without creating messy code. Besides that, it wraps the requests using a polyfill for ES6 new promise syntax. We can also intercept and cancel requests, and there is a built-in client-side protection against cross-site request forgery.

Next, there is a homepage that takes care of the basic requirements of the user which are all accessible with a single tap. For a mobile wallet the most important information are the basic details of the user, the balance, associated bank details, and transaction history of the user. All of this information can be easily retrieved using secure API connection’s in an online scenario. For the offline mode, these details are equally important and need to be displayed for the user to continue using the application. These details are stored in an offline environment using asynchronous storage. Asynchronous Storage is a storage system that developers can use and save data in the form of key-value pairs which is very similar to a browser’s local storage API. It alleviates the need to rely on heavy duty servers and external databases. This form of storage does not delay the loading of the app and can be quicker than simple data fetching. The key advantage of using asynchronous Storage over cookies and cache is that it limits the amount of data traveling back-and-forth to the server ensuring low latency.

The user can then navigate to the transaction page from the homepage. The transaction between two users is the most important part of the app and security must be a top priority in such payment apps. Given the already existing safety issues of API calls these days, we wanted to create a more secure channel/tunnel where only the concerned client and server can interact. With recent security issues, writing information using API calls can be a risk with hacker’s being able to modify POST requests. The client first tries to establish a Web-Socket connection if possible, and if not, the app initiates an offline mode transaction. In the situation where the customer has no internet connection, identified by NetInfo we make use of the asynchronous storage that...
react native has to offer to store a logged users information until the customer gets back online. After the customer gets a stable internet connection, the data is picked up from the asynchronous Storage and a secure socket connection is made from the homepage and is connected to the server and the transaction continues as it would similar to the online scenario.

For concurrency, we tried incorporating MongoDB’s multi-granularity locking that allows operations to lock at the global, database, or collection level and allows for individual storage engines to implement their own concurrency control below the collection level. But for our system implementation, we needed a row-level lock rather than a table-level lock. A table-level/collection level lock would slow down the whole user experience. The pipeline would block off incoming transactions and cause a delay. Hence we have implemented a document level for the application, which locks only a particular user enabling other users to continue with their transactions. Once the connection is established, any kind of communication between the client and the server will be made through this socket connection. After the connection is established, the user can make a transaction by providing the details of the merchant and the amount that has to be transferred.

After the transaction is complete, the user is redirected to a transaction confirmation page, post which they are redirected to the homepage from the payment gateway. From the homepage, the user can either continue making new transactions or can close the application.

To cater to the users needs and at the same time not make the application too complex for common users, we have worked on a system which enhances the capability of the current system being used. The online-offline application developed would ensure that end users will be able to have a full-fledged mobile wallet application at all times irrespective of the strength of their internet connectivity. This is an advancement to the current mobile wallet system being used in the market and provides a suitable replacement for online applications.

**Results and Discussion**

Based on the above-mentioned methodology an application has been created. The application is fully functional in both online and offline modes and security measures have been applied for both.

The M-Commerce offline payment application opens to a login page that asks the concerned customer or merchant to enter their credentials i.e, username and password. For added security, all passwords are hashed and stored. This initial setup is done when the user is online. Once this is done the details are hashed and stored in the device’s asynchronous storage. The data from the asynchronous storage cannot be tampered with and is safe for the user. This is subsequently used for all offline payments and ensures the safety of user data.

On creating a POST request via the route [/customers] the username is matched with the database and if matched, a response is sent to the client indicating that a customer with those credentials already exists, else a new customer is created and the procedure for it is as follows. First, the password entered by the end-user is hashed using NodeJS’ bcrypt library that takes in two parameters, one, the password to be encrypted and two, the salt to be used to hash the password, and if specified as a number, a salt will be generated with the specified number of rounds. Thereafter the end-users balance, associated bank, and phone number are recorded into the collection.

The first page of the app is the login page as seen in Fig. 3. Here, the user enters their credentials that have been stored in the database. The entered credentials are matched with the database and if they are correct. The login works through the routes set up through
Upon successful authentication, the end-user logs in and are redirected to the client’s homepage where the client would be able to see their remaining balance, their associated bank, and a history of the last five transactions made by them via this application. In the case the customer has a stable internet connection, clicking on the make payment button will take the user to the transaction page that is when a socket connection is established with the server, and once this connection is established, any other communication between the client and the server will take place through this socket. We have also coded an additional document level lock for Mongo DB, which ensures that multiple users can make a transaction at a given time without locking the whole collection. It works by locking the rows returned by a selection query, such that other transactions trying to access those rows are forced to wait for the transaction that locked the rows to finish. These other transactions are effectively put into a queue based on when they tried to read the value of the locked rows. The application takes care of concurrency by ensuring that a document/row-level lock is created so that the application does not contain any write/write and read/write conflicts. Once the server-side processing of the transaction is complete, the status of the transaction is sent back to the client through the same socket.

Figure 4 shows the Homepage of the App. The client details such as Username, associated bank account, and the current balance are displayed on this page. Along with such details, the transaction history of the client is displayed as well, which shows the last five transactions of the user, green depicting incoming and red depicting outgoing.

Each component has several “lifecycle methods” that we can override to run code at particular times in the process. We have used the component mounting life cycle method in our project. Mounting is the phase in which our React component will mount on the DOM i.e., is created and inserted into the DOM. This phase comes onto the scene after the initialization phase is completed. In this phase, our component renders for the first time. Once the user has entered valid credentials and login is successful, the App moves onto the homepage. Here, the componentWillMount() method is used which is called just before the component is mounted onto the DOM or the render method is called. Once this method is called, all the client details and the transaction history is queried from the database. React components are stacked on top of the other, for example, the homepage above the login and the payment above the homepage, which makes updating these components difficult as they only get updated with ReactDOM.render. This is why we added an additional listener (div focus) which calls the render method when we come back to our homepage, making sure that only the updated values are displayed. The .focus() method tells the app which screen is being acted on, which is helpful to find out which page the app is currently on. By exploiting that, the app can re-render when a payment is completed.

Moving on to the offline mode payments, we have made use of React Async Storage which is an unencrypted, asynchronous, persistent, and key-value storage system that is global to the app which is better than Local Storage. Here, the associated bank details and the balance amount is stored, which will be modified after a transaction. This data cannot be modified by the user at any time and is completely safe and secure. Hence offline transactions can be achieved without the fear of theft. We have made the use of a community library in react native called ’Netinfo’ which allows us to get information about connection type and connection quality. This helps us when the mobile gets a stable internet connection which the app then checks for using the Netinfo
library, after which, the last offline mode transaction can be updated online. This is done from our homepage itself and another socket connection is initiated through this page, which is then used to update the balance info.

Figure 5 is the transaction page of our application. When a customer wants to pay the merchant, the former will enter the transaction amount and the merchant’s phone number and thereafter if sufficient balance is available i.e., post-transaction the wallet value should be greater than zero, a successful transaction will have taken place and the customer will have redirected to the home page where their remaining balance and transaction history will be displayed. All these transactions are taken care of by NodeJS’ socket.io library which enables real-time, bidirectional, and event-based communication between the customer and the merchant. Web Socket is a communication protocol that provides a full-duplex and low-latency channel between the client and the server i.e. customer and merchant. Using Web Sockets, a developer can come up with a function that works consistently across all the platforms. Web Socket represents a single TCP socket connection, thus eliminating the problem of connection limitation. In the case of REST architecture, the client and server can be implemented independently, without knowing each other. This client/server paradigm has lots of benefits with it i.e., the client-side code can be altered at any time, without the server getting affected. The different clients having REST interfaces can hit the endpoints at the same time and receive the same response. Also, one other feature is statelessness. A server does not need to know which state the client is in, and the same holds true for the client. All in all, when it comes to security and information assurance it is necessary to ensure that the transactions or any such confidential information are passed through a secure socket only so that the chance of an intrusion attack or a network vulnerability is minimized.

Figure 6 shows the homepage of the application in an offline environment which is very similar to that of Fig. 4, the key difference being that the client will not be able to see the transaction history in the absence of a stable internet connection. On completion of a successful transaction, the customer is redirected to the 'Transaction Successful' page as seen in Fig. 7, and in the case of an unsuccessful transaction, the customer is redirected to the 'Transaction Failed' page as seen in Fig. 8. Post a transaction success or failure the customer will be redirected/navigated to the homepage.

The evaluation parameters, both in terms of cost and time analysis were extensively performed on our application. Over 1000 transactions between different customer and merchant accounts were done and the results from the analysis performed were better than the other wallet applications available in the market today. The time taken by our application for a given transaction was around 600ms on average which outperforms other benchmark wallet applications which are in the range of 700–800ms, this indicates that our proposed model works positively irrespective of any network-related traffic. The failures encountered so far were only due to business validation errors and not due to any technical faults. Business validation failures are those transactions that were initiated with the customer having insufficient balance and/or those where the customer gave an incorrect recipient for the transaction. The application developed has no technical issues seen thus far and even when a high density of transactions are observed at the same time, the application does not time out. This application is horizontally scalable and for industrial usage, additional nodes can be added for load management.

**Conclusion**

In this article we present a novel prototype of an offline wallet payment using React Native as our Frontend, NodeJS as our Backend and MongoDB as our database. The main idea of this prototype application was to handle and prevent
intrusion attacks and vulnerabilities, to ensure this we used socket.io library and made sure all our API calls were safe. For information assurance and security, we have made use of the bcrypt module, where we hash and store all passwords to our database, thereby reducing the chance of threats to an end user’s wallet. To establish concurrency and to avoid delay we implemented our own document level lock. Furthermore, for the offline mode of our prototype, the application makes use of asynchronous storage which is a global storage system which we used to store the transaction amount and bank details. This idea is very useful in day to day activities where a user may not always get a stable connection and would require to make transactions. This application allows users to continue using mobile payments in any scenario as mobile payments have bloomed in recent years. The internet barrier will allow several merchants and users to start using mobile payments. Security enhancements in the online mode has also been improved to create an absolutely safe environment for a user. At present the M-Commerce wallet application developed is centralised, the future scope of this project would be to make it a decentralised online-offline wallet payment system.

Fig. 6 Offline homepage

Fig. 7 Transaction successful
Declarations

Funding  There is no funding required for this project.

Conflict of Interest  The authors declare that they have no conflict of interest.

References

1. Jayasinghe Danushka, et al. Extending EMV tokenised payments to offline-environments. In: 2016 IEEE Trustcom/BigDataSE/ISPA. 2016;443–50. https://doi.org/10.1109/TrustCom.2016.60995.
2. San Aye Mi, Sathitwiriyawong Chanboon. "Privacy-preserving offline mobile payment protocol based on NFC". In: 2016 International Computer Science and Engineering Conference (ICSEC). 2016, pp. 1–5. https://doi.org/10.1109/ICSEC.2016.7859939.
3. Kiran N. Chitra, Narendra Kumar G. "Building robust m-commerce payment system on offline wireless network". In: 2011 Fifth IEEE Interna-tional Conference on Advanced Telecommunication Systems and Networks (ANTS). 2011, pp. 1–3. https://doi.org/10.1109/ANTS.2011.6163664.
4. He Yongqiang, Shi Yanrong, He Aixiang. "Research on online payment mode of e-commerce". In: 2010 IEEE International Conference on Software Engineering and Service Sciences. 2010, pp. 312–15. https://doi.org/10.1109/ICSESS.2010.5552440.
5. Luo Jia-Ning, Yang Ming-Hour. "Offline transferable E-cash mechanism". In: 2018 IEEE Conference on Dependable and Secure Computing (DSC). 2018, pp. 1–2. https://doi.org/10.1109/DESEC.2018.8625136.
6. Zhao Hao, Muftic Sead. "The concept of Secure Mobile Wallet". In: 2011 World Congress on Internet Security (WorldCIS-2011). 2011, pp. 54–58. https://doi.org/10.1109/WorldCIS17046.2011.5749882.
7. Kreft Heinz, Adi Wael. "Wallet based E-cash system for secured multi-hop cash exchange". In: 2008 3rd International Conference on Information and Communication Technologies: From Theory to Applications. 2008, pp. 1–5. https://doi.org/10.1109/ICTTA.2008.4529987.
8. Hassan Md Arif, Shukur Zarina. "Review of digital wallet requirements". In: 2019 International Conference on Cybersecurity (ICoCSec).2019, pp. 43–48.https://doi.org/10.1109/ICoCSec47621.2019.8970996.
9. Tso Raylin, Lin Chen-Yi. "An off-line mobile payment protocol providing double-spending detection". In: 2017 31st International Conference on Advanced Information Networking and Applications Workshops (WAINA). 2017, pp. 570–75.https://doi.org/10.1109/WAINA.2017.56.
10. Khan Burhan Ul Islam, et al. "Offline OTP based solution for Secure Internet Banking Access". In: 2018 IEEE Conference on e-Learning, e-Management and e-Services (IC3e). 2018, pp. 167–72.https://doi.org/10.1109/IC3e.2018.8632643.
11. Li Songnong, et al. "Research on offline transaction model in mobile payment system". In: Frontier Computing. Ed. by Jason C. Hung, Neil Y. Yen, and Lin Hui. Singapore: Springer Singapore, 2019, pp. 1815–20. ISBN: 978-981-3-3648-5.
12. Solat Siamak (2017) "Security of electronic payment systems: A comprehensive survey". arXiv. 1701.04556
13. Liu Wenzheng, Wang Xiaofeng, Peng Wei. "State of the Art: Secure Mobile Payment". In: IEEE Access8 2020, pp. 13898–914. https://doi.org/10.1109/ACCESS.2019.2963480.
14. Gupta Rahul, Kapoor Cheshtha, Yadav Jayesh. "Acceptance towards digital payments and improvements in cashless payment ecosystem". In: 2020 International Conference for Emerging Technology (INCEIT). 2020, pp. 1–9. https://doi.org/10.1109/INCEIT49848.2020.9154024.
15. Singh Jay P. "Offline transactions functionality in eWallets". In: Indian journal of science and technology10. 2017, pp. 1–4.
16. Samid Gideon, Wallet BitMint Hard. "Digital payment without network communication: No Internet, yet Sustained Payment Regimen between Randomness-Verifiable Hard Wallets". In: 2020 IEEE International IOT, Electronics and Mechatronics Conference (IEMTRONICS). 2020, pp. 1–7. https://doi.org/10.1109/IEMTRONICS51293.2020.9216456.
17. Rahat Abdullah, et al. "An offline electronic payment system based on untraceable blind signature scheme". In: KSIII transactions on internet and information systems 11. 2017, pp. 2628–45.https://doi.org/10.3837/tiis.2017.05.018.
18. Pal Abhipsa, et al. "Is the convenience worth the risk? An investigation of mobile payment usage". In: Information systems frontiers 23.4. 2021, pp. 941–61. https://doi.org/10.1007/s10796-020-10070.
19. Ahmed Waqas , et al. "Security in next generation mobile payment systems: A comprehensive survey". In: IEEE Access9. 2021, pp. 115932–950.
20. Wang Yong, Hahn Christen, Sutrave Kruttika. "Mobile payment security, threats, and challenges". In: 2016 Second International

Fig. 8  Transaction failed
Conference on Mobile and Secure Services (MobiSecServ). 2016, pp. 1–5. https://doi.org/10.1109/MOBISECSEVR.2016.7440226.

21. Hassan Md Arif, et al. "A review on electronic payments security". In: Symmetry 12.8. 2020; ISSN 2073–8994. https://doi.org/10.3390/sym12081344.

22. Abdullah, Khan Mohammed Naved. "Determining mobile payment adoption: A systematic literature search and bibliometric analysis". In: Cogent Business & Management 8.1 (2021). Ed. by Elena Kostadinova, p. 1893245. https://doi.org/10.1080/23311975.2021.1893245.

23. Canara Bank, Parvathi S. The impact of pandemic on digital payments in India. ResearchGate; 2021.

Publisher’s Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.