Digital Signature

A PROJECT REPORT

for

Mini Project (KCA353)

Session (2023-24)

**Submitted By**

**Akarsh Mishra**

**University Roll No- 2200290140016**

#### Submitted in partial fulfillment of the

#### Requirements for the Degree of

MASTER OF COMPUTER APPLICATIONS

#### Under the Supervision of

**Ms. Divya Singhal**

**(ASSISTANT PROFESSOR)**

### Submitted to

**DEPARTMENT OF COMPUTER APPLICATIONS**

#### KIET Group of Institutions, Ghaziabad

#### Uttar Pradesh-201206

**(March 2024)**

**CERTIFICATE**

Certified that **Akarsh Mishra 220029014008066** has carried out the project work having “**Digital Signature**” (**Mini Project-KCA353**) for **Master of Computer Application** from Dr. A.P.J. Abdul Kalam Technical University (AKTU**)** (formerly UPTU), Lucknow under my supervision. The project report embodies original work, and studies are carried out by the student himself/herself and the contents of the project report do not form the basis for the award of any other degree to the candidate or to anybody else from this or any other University/Institution.

**Date:**

**Akarsh Mishra**

**2200290140016**

This is to certify that the above statement made by the candidate is correct to the best of my knowledge.

**Date:**

**Ms. Divya Singhal Dr. Arun Tripathi**

**Assistant Professor Head**

**Department of Computer Applications Department of Computer Applications KIET Group of Institutions, Ghaziabad KIET Group of Institutions, Ghaziabad**

**DIGITAL SIGNATURE**

**AKARSH MISHRA**

**ABSTRACT**

Elliptic Curve Digital Signature Algorithms (ECDSA) have recently come into strong consideration, particularly by the standards developers, as alternatives to established standard cryptosystems such as the integer factorization cryptosystems and the cryptosystems based on the discrete logarithm problem. Crypto algorithms are always the most important core tool in security applications. The elliptic curve-based digital signature algorithms were implemented in this study using the open-source software from GNU's Not Unix (GNU) Compiler collection version 3.4.4-1 (2005). The introduction of Cygwin version 1.5.18-1 (2005) released into this study, also enable users of Microsoft windows to make use of the software. ECDSA was examined and the main reason for the attractiveness of elliptic curve cryptography was brought out in the fact that there is no sub-exponential algorithm so far known to solve the elliptic curve discrete logarithm problem on a properly chosen elliptic curve. Hence, it takes full exponential time to solve while the best algorithms known for solving the underlying integer factorization and discrete logarithm problem both take sub-exponential time. The ECDSA has a smaller key size, which leads to faster computation time and reduction in processing power, storage space and bandwidth. This makes ECDSA ideal for constrained environments such as pagers, cellular phones, and smart cards.

# ACKNOWLEDGEMENT

Success in life is never attained single-handedly. My deepest gratitude goes to my project supervisor, **Ms. Divya Singhal,** for their guidance, help and encouragement throughout my research work. Their enlightening ideas, comments, and suggestions.

Words are not enough to express my gratitude to **Dr. Arun Kumar Tripathi, Professor and Head, Department of Computer Applications**, for his insightful comments and administrative help at various occasions.

Fortunately, I have many understanding friends, who have helped me a lot in many critical conditions. Finally, my sincere thanks go to my family members and all those who have directly and indirectly provided me with moral support and other kinds of help. Without their support, completion of this work would not have been possible in time. They keep my life filled with enjoyment and happiness.

**Akarsh Mishra**

**(2200290140016)**

# List of Chapters

|  |
| --- |
| **Certificate i** |
| **Abstract ii** |
| **Acknowledgements iii** |
| **Table of Content iv** |
| **Chapter 1 – Introduction** |
| 1.1 Project description 1 |
| 1.2 Literature Review 3 |
| 1.3 Privacy and Security 4 |
| 1.4 Software & Hardware requirements 7 |
| 1.5 Functional Requirements 8  1.6 Nonfunctional Requirements 9 |
| **Chapter 2 Feasibility Study** |
| 2.1. Technical Feasibility 10 |
| 2.2. Financial Feasibility 11  2.3. Operational Feasibility 12 |
| **Chapter 3 Database Design** |
| 3.1 Waterfall Model 13 |
| 3.2 Requirement Gathering & Analysis 14 |
| 3.3 ER Diagram 15 |
| 3.4 Use Case Diagram 17 |
| 3.5 Activity Diagram 18  **Chapter 4 Code Preview**  4.1. Discussion and Sample Code 19  4.2. Predetermined keys 20  4.3. Testing 20  **Chapter 5 Conclusion & Future Work** 23  Bibliography 24 |

**CHAPTER – 1**

**INTRODUCTION**

#### PROJECT DESCRIPTION

Cryptography is a multifaceted field that revolves around securing communication in the presence of potential adversaries. Its primary goal is to develop and analyze protocols that safeguard information shared between entities, ensuring that malicious third parties cannot access it. This practice is essential for various aspects of information security.

One of the central principles of modern cryptography is data confidentiality, which involves restricting access to information through confidentiality agreements or rules. Data integrity is another critical aspect, ensuring that data remains accurate and consistent throughout its lifecycle. Authentication is the process of verifying that the claimed user corresponds to the data, while non-repudiation ensures that parties cannot deny the authenticity of their signature or message sending in contracts or communications.

Digital signatures serve as the cornerstone of message authentication in the realm of public-key cryptography. They bind individuals or entities to digital data, allowing independent verification by receivers and third parties. In practical scenarios, recipients require assurance that a message originates from the claimed sender and cannot be repudiated, making digital signatures vital for businesses where disputes over exchanged data are common. Digital signatures offer higher security than other electronic signature forms, enhancing transparency in online interactions and fostering trust among stakeholders. These signatures rely on mathematical techniques to validate the authenticity and integrity of digital content, acting as a more secure alternative to handwritten signatures or seals. They address the challenges of tampering and impersonation in digital communications.

Digital signatures also provide evidence of origin, identity, and the status of electronic documents, transactions, or messages. In numerous countries, including the United States, digital signatures hold the same legal weight as traditional handwritten signatures on documents. Blockchain technology, known for underpinning cryptocurrencies like Bitcoin, Ethereum, and Litecoin, relies on public-key cryptography to protect user accounts from unauthorized access. Public and private keys enable encryption and verification of messages, ensuring message authority and detecting alterations caused by adversaries.

In summary, cryptography is instrumental in securing communication and data in the digital age, with digital signatures playing a crucial role in verifying authenticity and integrity. Moreover, blockchain technology relies on cryptographic principles to safeguard user accounts and maintain the integrity of transactions in the realm of cryptocurrencies.

# LITERATURE REVIEW

The Digital Signature Project is a comprehensive endeavor aimed at harnessing the power of cryptographic technology to enhance the security and trustworthiness of digital communications and transactions. This project delves into the multifaceted world of cryptography, emphasizing the pivotal role of digital signatures to and integrity of digital data. By exploring the validate the authenticity principles of data confidentiality, integrity, authentication, and non-repudiation, this initiative seeks to establish a robust framework for secure communication in the presence of potential adversaries. Furthermore, the project extends its focus to the application of digital signatures in real-world scenarios, particularly in business applications, where the prevention of disputes over exchanged data is paramount. Additionally, it explores the integration of digital signatures into emerging technologies like blockchain, showcasing their indispensable role in safeguarding accounts and transactions in the digital realm.

The history of digital signatures spans several decades, marked by technological milestones and the evolution of cryptographic techniques. In the 1970s, public-key cryptography, introduced by Diffie and Hellman, laid the foundation. The RSA algorithm, developed by Rivest, Shamir, and Adleman in 1977, became pivotal for digital signatures.

The 1980s saw the emergence of the Digital Signature Standard (DSS) by NIST, with the adoption of the Digital Signature Algorithm (DSA) in 1991. By the 1990s, digital signatures gained recognition in governments and businesses, leading to legislation and legal recognition in various countries.

The early 2000s witnessed widespread adoption, particularly in e-commerce, online banking, and secure email communication. Standardization efforts resulted in XML Digital Signatures (XMLD Sig) in 2000, providing a universal format. Advancements in cryptographic techniques, such as Elliptic Curve Cryptography (ECC), improved efficiency and security. In the 2010s, blockchain technology integrated digital signatures, enhancing trust and integrity in transactions.

Today, digital signatures are legally binding in many countries and industries, serving as a standard for secure authentication, document signing, and data integrity. This historical journey reflects the continuous pursuit of secure methods to authenticate digital content and transactions, shaping the modern digital landscape.

### 1.3. Privacy and Security

Privacy and security are critical aspects of digital signatures, which are cryptographic techniques used to verify the authenticity and integrity of digital messages or documents. Here are some key considerations regarding privacy and security in digital signatures:

**Authentication:** Digital signatures provide a means to authenticate the origin of a message or document. The digital signature ensures that the sender is who they claim to be.

**Integrity:** Digital signatures also ensure the integrity of the message or document. Any alteration to the content of the message will result in the invalidation of the signature.

**Cryptographic Algorithms:**

The security of digital signatures relies on robust cryptographic algorithms. Commonly used algorithms include RSA (Rivest-Shamir-Adleman), DSA (Digital Signature Algorithm), and ECDSA (Elliptic Curve Digital Signature Algorithm).

It's crucial to use algorithms with sufficiently long key lengths to resist brute-force attacks.

**Key Management:**

The private key used for creating digital signatures must be kept confidential. Unauthorized access to the private key could compromise the security of the digital signature. Secure key management practices, such as using hardware security modules (HSMs) or secure key storage, should be implemented.

**Public Key Infrastructure (PKI):**

PKI is a framework that manages digital keys and certificates. It includes a Certificate Authority (CA) that verifies the identity of entities and issues digital certificates.

PKI helps establish a trust hierarchy, ensuring that the public key in a digital certificate corresponds to the private key held by the claimed entity.

**Secure Transmission:**

When transmitting digitally signed messages, ensure the use of secure communication protocols (e.g., TLS/SSL) to protect the message during transit.

Encryption of the entire communication channel adds an extra layer of security.

**Timestamping:**

Including a timestamp in the digital signature can provide evidence that the signature was created at a specific point in time. This can be crucial for legal and regulatory purposes.

**Revocation:**

Establish mechanisms for revoking digital certificates in case a private key is compromised or if the entity's status changes. Certificate Revocation Lists (CRLs) and Online Certificate Status Protocol (OCSP) are common methods for checking the validity of digital certificates.

**Legal and Regulatory Compliance:**

Ensure that your digital signature solution complies with relevant legal and regulatory requirements. Different jurisdictions may have specific laws governing the use of digital signatures.

**User Awareness and Training:**

Educate users about the importance of protecting their private keys, using secure passwords, and following best practices for digital signature usage.

By addressing these aspects, organizations can enhance both the privacy and security of their digital signature implementations. It's essential to stay informed about evolving cryptographic standards and best practices to adapt to emerging security threats.

**Impact on Social Relationships:**

The use of digital signatures can have various impacts on social relationships, both in personal and professional contexts. Here are some ways in which digital signatures can influence social relationships:

**Trust and Credibility:**

Digital signatures contribute to building trust and credibility in online interactions. The ability to verify the authenticity and integrity of digital documents enhances trust between individuals, businesses, and organizations.

**Convenience and Efficiency:**

Digital signatures streamline processes by eliminating the need for physical signatures and paperwork. This can lead to more efficient and convenient interactions, fostering positive relationships by saving time and resources.

**Global Collaboration:**

Digital signatures facilitate collaboration across geographical boundaries. People can sign documents and agreements without the need for physical presence, enabling global partnerships and interactions that may not have been feasible otherwise.

**Reduced Paper Dependency:**

The shift from traditional, paper-based signatures to digital signatures contributes to environmental sustainability. This eco-friendly approach can be appreciated in social circles where individuals prioritize environmentally conscious practices.

**Security and Privacy Concerns:**

While digital signatures enhance document security, concerns about privacy and data protection may arise. Individuals may be cautious about sharing personal information online or signing sensitive documents digitally, affecting their willingness to engage in certain interactions.

**Learning Curve and Technology Divide:**

The adoption of digital signatures may pose challenges for those who are not familiar with technology. A significant technological divide could impact social relationships, as some individuals may feel excluded or face difficulties in adapting to digital practices.

**Legal Recognition and Compliance:**

The legal recognition of digital signatures varies across jurisdictions. In regions where digital signatures have strong legal standing, individuals and businesses may feel more confident in engaging in digital transactions. In areas with less legal clarity, concerns about the enforceability of digitally signed documents may affect relationships.

**Communication Patterns:**

The ease of digitally signing documents may influence communication patterns. For instance, individuals may prefer virtual communication over face-to-face interactions for certain types of agreements, potentially affecting the personal and social aspects of relationships.

**Collaboration Tools and Platforms:**

The use of digital signatures is often integrated into collaboration tools and platforms. This integration can impact the way people collaborate and communicate within these platforms, influencing the dynamics of social relationships within professional networks.

**Adoption in Social Contracts:**

In personal relationships, digital signatures may be used for various agreements, such as shared documents, contracts, or even informal agreements. The adoption of these practices can reflect changing social norms and the evolution of interpersonal relationships.

In summary, the impact of digital signatures on social relationships is multifaceted, with both positive and potentially challenging aspects. It often depends on factors such as technological literacy, legal frameworks, and individual attitudes toward the use of digital tools in social interactions.

# 1.4. Software & Hardware requirements

1. Operating system- Windows 10
2. Programming language – Java
3. Java version “1.7.0\_25”
4. Database \_SQL
5. ECC library
6. Hash Function- SHA256withECDSA
7. ECC provider test
8. Java library- java. Security
9. Documentation tool - Ms Word
10. 64-bit operating system
11. x64-based processor
12. Intel(R) Core (TM) i3-4030U CPU @ 1.90GHz 1.90 GHz

# 1.5 FUNCTIONAL REQUIREMENTS

Functional requirements are specifications that describe the system's features, capabilities, and functionalities. These requirements outline what the system is expected to do and serve as the basis for system design and development. Here is a general structure and some examples of functional requirements:

**1. User Authentication:**

The system must provide a user authentication mechanism to ensure secure access.

* Users must be able to create an account.
* Users must be able to log in using a valid username and password.
* The system must support multi-factor authentication.

**2. Data Management:**

The system must handle the storage and retrieval of data efficiently. The system must support CRUD (Create, Read, Update, Delete) operations on user data. Data must be stored securely and retrievable based on user permissions.

**3. Search Functionality:**

The system must provide a search feature for users to find relevant information. Users must be able to search for data based on specified criteria. The search results must be displayed in a clear and organized manner.

**4. Notification System:**

The system must notify users about important events or updates. Users should receive email notifications for account-related activities. In-app notifications must be provided for real-time updates.

**5. Reporting and Analytics:**

The system must generate reports and provide analytics for decision-making. Users must be able to generate predefined reports. The system should support customizable analytics dashboards.

**6. Workflow Management:**

The system must support defined workflows for various processes. Users should be guided through a step-by-step process for specific tasks. Workflow customization options must be available for administrators.

**7. Integration with Third-Party Systems:**

The system must integrate with external systems as needed. The system must be able to exchange data with external APIs. Integration with external authentication services should be supported.

**8. Security and Access Control:**

The system must ensure the security of data and control access appropriately. Data transmission must be encrypted using HTTPS. Access to certain features or data should be restricted based on user roles.

**9. User Interface (UI) Design:**

The system must provide an intuitive and user-friendly interface. The UI must be responsive and accessible on various devices. Consistent design elements and navigation should be maintained.

**10. Performance:**

The system must perform efficiently under expected workloads. Response times for common operations should be within acceptable limits. The system must handle a specified number of concurrent users. These are just examples, and the specific functional requirements will vary based on the nature of the system being developed. Functional requirements help guide the development team in building a system that meets the users' needs and expectations

# 1.6 NON-FUNCTIONAL REQUIREMENTS

Non-functional requirements define the qualities or attributes that describe the overall characteristics of a system. Unlike functional requirements that specify what the system should do, non-functional requirements outline how well the system should perform its functions. Here are examples of non-functional requirements:

**1. Performance:**

The system must respond to user requests within 2 seconds. This non-functional requirement defines the expected response time for user interactions to ensure a responsive user experience.

**2. Scalability:**

The system must handle a 20% increase in concurrent users without degradation in performance. Specifies the system's ability to handle increased loads, ensuring scalability to accommodate growing user numbers.

**3. Reliability:**

The system must have an uptime of at least 99.9%. Defines the reliability expectation, indicating the acceptable downtime for maintenance or unexpected issues.

**4. Availability:**

The system should be available 24/7, except during scheduled maintenance windows.

Describes the expected availability of the system, including any planned downtime for maintenance.

**5. Security:**

User data must be encrypted during transmission and storage using industry-standard encryption algorithms. Specifies security measures to protect user data, both during communication and while stored in the system.

**6. Usability:**

The system must adhere to WCAG 2.0 accessibility standards. Ensures that the system is designed to be accessible to users with disabilities, contributing to overall usability.

**7. Maintainability:**

Code changes should be deployable with minimal downtime, and the system should support easy updates. Describes the ease with which the system can be maintained, including deploy ability and support for updates.

**8. Compatibility:**

The system must be compatible with the latest versions of major web browsers (Chrome, Firefox, Safari, Edge). Ensures that the system functions correctly across different browsers, contributing to a consistent user experience.

**9. Compliance:**

The system must comply with relevant industry regulations and data protection laws.

Specifies that the system should adhere to legal and regulatory requirements, ensuring ethical and legal use.

# CHAPTER-2

### FEASIBILITY STUDY

A feasibility study is an essential step in the early stages of project planning, helping stakeholders assess the viability and potential success of a proposed project. It typically includes an analysis of various aspects, such as technical, economic, legal, operational, and scheduling considerations. Below are key components typically covered in a feasibility study:

**2.1. Technical Feasibility:**

In evaluating the technical feasibility of the project, it's imperative to embark on a comprehensive examination encompassing several key aspects. Firstly, a meticulous technology assessment should be conducted to thoroughly analyze the landscape of available technologies and their suitability in achieving the project's objectives. This assessment should delve into factors such as the scalability, reliability, and compatibility of each technology with the project's specific requirements.

Furthermore, it is essential to identify and scrutinize the infrastructure needs essential for the seamless execution of the project. This involves assessing the existing infrastructure and determining any gaps that need to be addressed to support the implementation of the chosen technical solutions effectively. Considerations should include network capabilities, hardware requirements, data storage and processing resources, and any specialized equipment or facilities needed.

Moreover, a detailed examination of the practicality of implementing the necessary technical solutions within the project framework is paramount. This involves assessing factors such as cost-effectiveness, time constraints, regulatory compliance, and potential risks associated with the deployment of technology. Additionally, considerations should be made regarding the expertise and resources required for development, integration, and ongoing maintenance of the technical solutions.

By meticulously scrutinizing these aspects, the project can ensure that it is firmly grounded in a realistic understanding of its technical demands and capabilities. This comprehensive approach significantly enhances the project's chances of success by mitigating risks, optimizing resource allocation, and facilitating informed decision-making throughout its lifecycle.

**2.4. Financial Feasibility:**

This section involves a thorough and detailed examination of the project's financial landscape, aiming to provide a comprehensive understanding of its economic viability and sustainability. It encompasses a range of critical analyses, including cost estimation, revenue projections, return on investment (ROI) calculations, and financial risk assessment.

To begin with, cost estimation entails meticulously identifying and quantifying all expenses associated with the project across its lifecycle. This involves not only direct costs such as materials, labor, and equipment but also indirect costs like administrative overhead, regulatory compliance, and potential contingencies. Moreover, accounting for any potential fluctuations in costs, inflation rates, or market conditions adds depth to the estimation process.

On the revenue side, projecting potential income streams involves forecasting sales, subscriptions, or other revenue sources relevant to the project. This forecasting should consider market demand, competitive landscape, pricing strategies, and any external factors that may impact revenue generation over time. Sensitivity analysis can also be employed to assess the robustness of revenue projections under different scenarios.

Calculating the return on investment (ROI) is a pivotal step in evaluating the profitability and viability of the project. This involves comparing the expected financial gains against the total investment required, taking into account both initial capital outlay and ongoing operational expenses. ROI analysis provides stakeholders with valuable insights into the project's potential financial performance and helps in decision-making regarding resource allocation and prioritization.

Furthermore, conducting a comprehensive financial risk assessment is essential to identify and mitigate potential threats to the project's financial success. This entails evaluating various risk factors such as market volatility, regulatory changes, technological disruptions, and financial constraints. Risk mitigation strategies can then be developed to address these challenges and safeguard the project's financial sustainability.

By scrutinizing these aspects in detail, the project can gain valuable insights into its financial feasibility and make well-informed decisions to optimize its financial performance and ensure sustainable success over the long term.

**2.5. Operational Feasibility:**

The operational feasibility analysis within this section represents a meticulous examination aimed at assessing the practicality and viability of integrating the proposed system within the organization's existing framework. This comprehensive evaluation encompasses various critical considerations that are pivotal for ensuring the successful adoption and utilization of the proposed system.

First and foremost, it involves assessing the impact of the proposed system on the organization's current processes and workflows. This entails identifying potential synergies and disruptions that may arise from the introduction of the new system, as well as evaluating how well it aligns with the organization's strategic objectives and operational requirements.

Furthermore, the analysis delves into the availability of resources necessary for implementing and sustaining the proposed system. This includes evaluating human resources in terms of skillsets, expertise, and availability, as well as financial resources required for procurement, implementation, and ongoing maintenance. Additionally, an assessment of technological resources such as infrastructure, software compatibility, and scalability is essential to ensure seamless integration and optimal performance of the system.

Moreover, evaluating the organization's readiness for change is imperative for anticipating and addressing potential challenges associated with transitioning to the new system. This involves gauging stakeholder buy-in and support at various levels of the organization, as well as identifying training requirements to enhance employees' proficiency with the new system. Cultural alignment is also a crucial aspect, as it determines the organization's receptiveness to change and its ability to adapt to new processes and technologies.

By scrutinizing these factors comprehensively, the project can ascertain the feasibility of implementing the proposed system and make informed decisions to ensure its seamless integration and operational success within the organization. This detailed analysis enables stakeholders to identify potential risks and challenges proactively and develop strategies to mitigate them, thereby enhancing the likelihood of a successful implementation and realizing the intended benefits of the new system.

# CHAPTER – 3

**DATABASE DESIGN**

### Waterfall Model

The waterfall model is a well-known structured methodology for software development. The whole process of system development is divided into distinct phases. The model has been introduced in the 1970s. Every phase has a unique output.

It was the first SDLC model to be used widely. So that, sometimes it is referred to Waterfall by SDLC. The waterfall model is used when the system requirements are well known, technology is understood, and the system is a new version of an existing product (Dennis, Wixom and Roth, 2012)

Mainly there are six phases in the Waterfall model. If there is a problem faced in any phase of the cycle, the system goes to the previous phase. The phases of Waterfall method are:

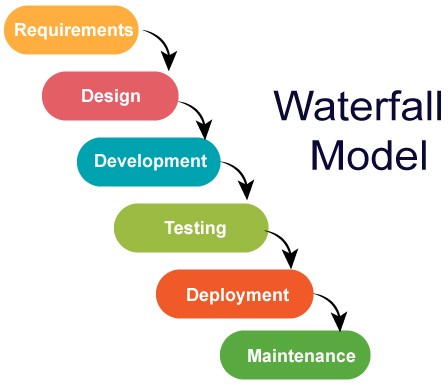


Fig. 3.1 Waterfall Model

### Requirements gathering & analysis

In this Phase, all possible requirements of the system are captured and documented in a requirement specification doc.

### System Design:

### The requirements documented in the previous phase are studied in this phase and the system design is prepared.

### Implementation:

With inputs from system design, the system is developed in several units. Then the units are tested.

### Integration & Testing:

The units of the program developed in the previous phase are integrated into a system. Then the whole system is tested.

### Deployment of the system:

When all kinds of testing is done, the product is deployed in the customer environment.

### Maintenance:

There are some issues which are found in the client environment. Patches are released to fix those issues.

### ER DIAGRAM

An Entity-Relationship (ER) diagram is a visual representation of the relationships between entities in a system. However, it might not be the most suitable diagram to represent the concept of a digital signature, which is a cryptographic concept used for verifying the authenticity and integrity of digital messages or documents.

Instead, I can provide you with a high-level description of the components involved in a digital signature system:

**User Entity:**

Represents an individual or entity involved in the digital signature process.

**Digital Signature Entity:**

Represents the digital signature itself, which is a cryptographic value generated based on the content of a message or document.

**Message or Document Entity:**

Represents the digital content that needs to be signed.

**Private Key Entity:**

Represents the private key of the user, which is used to generate the digital signature.

**Public Key Entity:**

Represents the public key of the user, which is used to verify the digital signature.

**Key Pair Relationship:**

Connects the User Entity with the Private Key and Public Key Entities, indicating that each user has a unique pair of private and public keys.

**Signing Relationship:**

Connects the User Entity with the Digital Signature and Message or Document Entities, indicating that a user signs a message or document using their private key.

**Verification Relationship:**

Connects the User Entity with the Digital Signature and Message or Document Entities, indicating that a user can verify the authenticity and integrity of a message or document using the sender's public key and the digital signature.

Please note that this is a conceptual representation, and the actual implementation might involve more details and cryptographic operations. Additionally, specific details may vary depending on the digital signature algorithm used (e.g., RSA, DSA, ECDSA).

***User Entity has the following attributes.***

**User ID (or Username):**

A unique identifier for each user in the system.

**Full Name:**

The full name of the user or the name of the entity.

**Email Address:**

Contact information associated with the user.

**Password:**

For security reasons, there might be a password attribute to protect the user's private key.

**Private Key:**

A unique cryptographic key known only to the user for generating digital signatures.

**Public Key:**

A cryptographic key derived from the private key and shared publicly for signature verification.

**Date of Registration:**

The date when the user registered in the system.

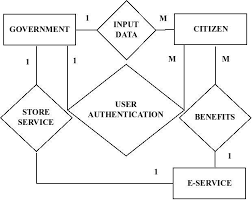


Fig. 3.2 ER Diagram

**3.4 USE CASE DIAGRAM**

A Use Case Diagram provides a high-level view of the functionality provided by a system and the actors (users or external systems) interacting with the system. In the context of a digital signature system, the main actors typically include the User and the System itself. Here's a simplified Use Case Diagram for a digital signature system:

**User:**

Represents an individual or entity that interacts with the digital signature system.

**System:**

Represents the digital signature system itself.

**Use Cases:**

**Generate Digital Signature:**

The user initiates the process of generating a digital signature for a specific message.

**Sign Document:**

The user uses their private key to sign a document, producing a digital signature.

**Verify Digital Signature:**

The user or another party initiates the process of verifying the authenticity and integrity of a document using the digital signature and the sender's public key.

**Manage Key Pair:**

The user manages their key pair, which includes actions like key generation, key update, or key deletion.

**View Signature History:**

The user can view a history of digital signatures they have generated or interacted with.

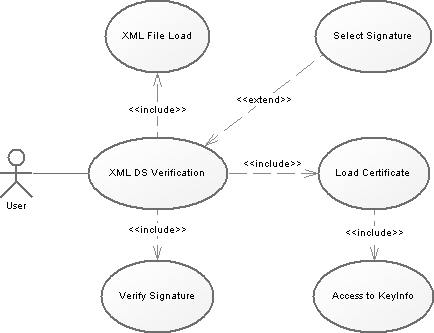


Fig. 3.3 Use Case Diagram

# 3.5. ACTIVITY DIAGRAM

An Activity Diagram is a behavioral diagram. It depicts the behavior of a system. Its primary use is to depict the dynamic aspects of a system. The dynamic aspect of a system specifies how the system operates to attain its function.

It is basically a flowchart to represent the flow from one activity to another activity. Activity Diagrams are not exactly flowcharts as they have some additional capabilities including branching, parallel flow, etc.

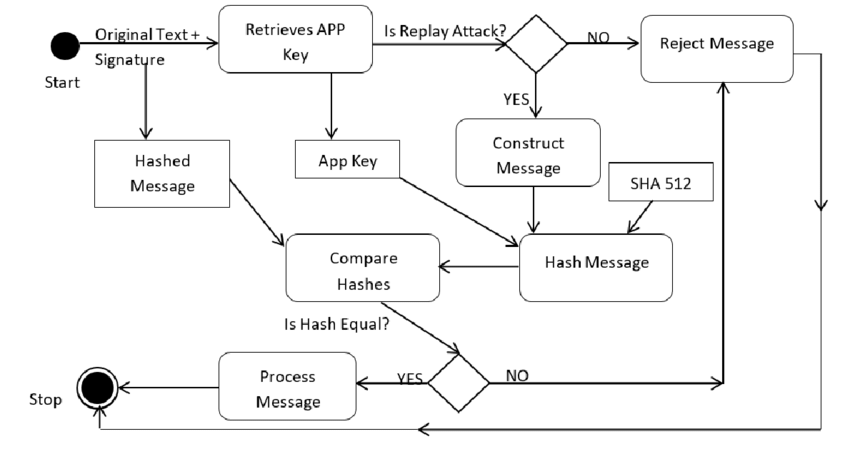


Fig 3.5 Activity Diagram

**Chapter 4**

**CODE PREVIEW**

**4.1. Discussion and Sample Code**

**The program named RSA09**

Because this program uses predetermined keys that were designed for a block size of 12, it will work correctly only for a block size of 12.

**The purpose of the program**

The purpose of this program is to illustrate the signing of an unencrypted message using an encrypted message digest as the digital signature, along with the later verification of the signature by the receiver of the message.

**Alice and Bob**

Many of the comments in this program reflect the scenario of Alice and Bob, which is common in discussions of RSA.  Alice needs to send a *signed* unencrypted message to Bob.  The message needs to be signed so that Bob can be confident that the message that he received was actually sent by Alice and was not sent by someone pretending to be Alice.

Bob also needs to confirm the integrity of the message by confirming that the message was not modified during the transmission from Alice to Bob.

**Creating a digital signature**

Alice creates a digital signature by getting an SHA-1 message digest for her message.  Then she encodes and encrypts the digest using her private key and the RSA algorithm.  The encrypted digest becomes her digital signature.  She signs the message by appending the encrypted digest to the end of the message.  Bob can later decrypt the digital signature using Alice's public key.

**Separating the message from the signature**

Bob will need some way to separate the message from the digital signature.  Therefore, before sending the signed message to Bob, Alice appends four numeric characters to the end that specify the length of the original message in characters.  Then Alice sends the signed message to Bob.

Bob extracts the message length from the end of the signed message.  He uses this information to separate the original message from the digital signature.  Then he decrypts and decodes the digital signature using Alice's public key.

**What does Bob already know?**

Bob knows that the original message digest was 40 hexadecimal characters in length, because an SHA-1 message digest is always 160 bits or 20 bytes in length.  Two hexadecimal characters are required to represent each byte, resulting in 40 hexadecimal characters for a hexadecimal representation of an SHA-1message digest.

Bob also knows that any additional characters beyond the first 40 hexadecimal characters were the result of extending the message digest to make it possible to encrypt it using the RSA encryption algorithm.  Bob discards all but the first 40 characters from the decoded digital signature.

**Checking the message digest**

Then Bob computes an SHA-1 message digest for the extracted message and compares it with the first 40 characters of the decoded digital signature.  If they match, he concludes that the digital signature is valid.  If not, he concludes that the digital signature is invalid.

If the digital signature is valid, Bob has confirmed both the authenticity and the integrity of the message from Alice.  It could only have been sent by Alice *(or by someone having access to her private key).*  Furthermore, the message could not have been modified after Alice computed her message digest and created her digital signature.

This program allows the message to contain all of the ASCII characters from space (32) through ~ (126) inclusive.

**4.2. Predetermined keys**

This program uses predetermined values for the following:

* Alice's public key, e
* Alice's private key, d
* Alice's modulus operand, n

**4.3. Testing**

This program was tested using Sun's JDK 1.5 (BlueJ as java editor) and Win11.

This program produces the output shown in Figure 1. I will refer back to Figure 1 as I explain the code in this program.

**Algorithm 1:**

|  |
| --- |
| 1. Alice's keys:  2. e: 17  3. d: 279263220413  4. n: 951386374109  5. Block size: 12  6. Alice's msg: Hello Bob, how are you? I am going to go to the movie tonight. I plan to see Gone with the Wind. I really like Clark Gable. Would you like to go with me? I would really like to have some company. Your friend, Alice.  7. Alice's msg digest: 89CA87110E2063FAF4A364B77C18C3260AD47B8E  8. Alice's extended msg digest: 89CA87110E2063FAF4A364B77C18C3260AD47B8E00000000  9. Alice's encrypted digital signature: 511335051  5586806264156901277970003516182249744328267808527  91138100851742238493708680633911102408  10. Alice's signed msg: Hello Bob, how are you? I am going to go to the movie tonight. I plan to see Gone with the Wind. I really like Clark Gabl e. Would you like to go with me? I would really like to have some company. Your friend,  Alice.5  1133505155868062641569012779700035161822497443282  6780852791138100851742238493708680633911102408  11. Alice's signed msg with msg length appended: Hello Bob, how are you? I am going to go to the movie tonight. I plan to see Gone with the Wind. I really like Clark Gable. Would you like to go with me? I would really like to have some company. Your friend,  Alice. 51133505155868062641569012 7797000351618224974432826780852791138100851742238 4937086806339111024080220  12. Bob's calculated msg length: 220  13. Bob's extracted msg text: Hello Bob, how are you? I am going to go to the movie tonight. I pl an to see Gone with the Wind. I really like Clar k Gable. Would you like to go with me? I would really like to have some company. Your friend, A lice.  14. Bob's extracted extended digital signature: 51133505155868062641569012779700035161822497443282 6780852791138100851742238493708680633911102408  15. Bob's decoded extended digital signature: 89C  A87110E2063FAF4A364B77C18C3260AD47B8E00000000  16. Bob's decoded digital signature: 89CA87110E2063FAF4A364B77C18C3260AD47B8E  17. Bob's digest: 89CA87110E2063FAF4A364B77C18C3260AD47B8E  18. Bob's conclusion: Valid signature |

**CHAPTER 5**

**CONCLUSION & FUTURE WORK**

Systems based upon elliptic curves are an effective alternative to the RSA. Cryptosystems since they involved different mathematical approaches. Elliptic curve cryptosystems are reputed for robustness equivalent to RSA cryptosystems with shorter key length. Accordingly, elliptic curve cryptosystems are perfectly suitable for embedded systems, e.g., smart cards, documents in which memory and power of the processors are not sufficient to achieve computation as required by RSA cryptosystems.

However, ECC and RSA cryptosystems involved the use of keys for security. We will be Due to the major drawback of the symmetric algorithms; we therefore concluded the algorithm implementing the elliptic curves is less costly in terms of key length. This is the main reason why the ECC algorithm is increasingly becoming the preferred choice for embedded systems with very limited memory and computing power.

Considering Bandwidth saving, ECC offers considerable bandwidth savings over RSA and considering computational overheads, ECC offers Roughly 10 times than that of RSA can be saved [18]. Considering key sizes, System parameters and key pairs are shorter for the ECC than RSA and After the different results obtained at the level of the encryption and decryption and those obtained for the digital signature, we have deduced that the elliptic curve algorithms are more efficient than the one based on the RSA. Also, we found that the digital signature ECDSA is more efficient than the ECNR. This is why among the algorithms based on the elliptic curves, the ECDSA algorithm was adopted and published as an international standard in ANSI X9.623.

We proposed the first ECDLP based GSS which is more efficient and secure due to use of Elliptic Curve cryptography and can be used practically in the situations if there is a need to hide identity of Group of signers while any member can sign on behalf of Group. Our proposed scheme can also be used in Blockchain applications.

The query pertains to future developments in the realm of digital signatures. This likely involves exploring forthcoming advancements in cryptographic techniques, key management, and authentication methods to enhance the security, efficiency, and usability of digital signatures. Additionally, there may be discussions about integrating emerging technologies like blockchain or quantum-resistant cryptography to address potential vulnerabilities and ensure the long-term viability of digital signatures in an ever-evolving technological landscape.

**BIBLIOGRAPHY**

## **Web Resources**

1. [**www.java.sun.com-**](http://www.java.sun.com-)Official Java Website
2. [**www.java.sun.com/developer/onlineTraining/J2EE/Intro2/j2ee.html**](http://www.java.sun.com/developer/onlineTraining/J2EE/Intro2/j2ee.html)Training for J2EE
3. [**www.java.sun.com/j2se/1.4.2/docs/api/index.html**](http://www.java.sun.com/j2se/1.4.2/docs/api/index.html)J2SE Online Documentation from Sun
4. [**www.w3schools.com**](http://www.w3schools.com) JavaScript Tutorials

## **BOOKS**

1. **API DOCS –JAVA, J2EE, Java Mail, Java Servlets, JSPs**

By: Sun Microsystems

1. **Java2 - The Complete Reference(7TH Edition)**

By: Herbert Schildt

1. **JSP - The Complete Reference**

By: Philhanna

1. **Oracle 10g**

By: Ivan Baross

1. **Software Engineering**

By: Roger Pressman

1. **Head First Servlets & JSP**

By: Bryan Bashan, Kasthy Sierra & Bert Bate

## **Journal Articles**

1. <https://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.72.3151&rep=rep1&type=pdf>
2. Hussein\_2020\_IOP\_Conf.\_Ser.\_\_Mater.\_Sci.\_Eng.\_928\_0320221
3. <https://link.springer.com/10.1007%2F978-1-4419-5906-5_208>
4. [**https://en.wikipedia.org/wiki/Group\_signature#:~:text=A%20group%20signature%20scheme%20is,Eugene%20van%20Heyst%20in%201991**](https://en.wikipedia.org/wiki/Group_signature#:~:text=A%20group%20signature%20scheme%20is,Eugene%20van%20Heyst%20in%201991)**.**
5. **https://eprint.iacr.org/2005/325.pdf**