BAB III
METODE PENELITIAN

3.1 Identifikasi Masalah

Berdasarkan pemaparan pada kajian pustaka, peneliti tertarik untuk mengimplentasikan algoritma RSA dan SHA-256 pada aplikasi *e-voting*. Algoritma RSA digunakan untuk proses merahasiakan pilihan suara sedangkan algoritma SHA-256 akan digunakan untuk proses keabsahan dari pilihan suara. Aplikasi ini akan berbasis web yang dibangun dengan *Javascript*.

Proses yang pertama dilakukan merupakan pembangkitan kunci yang akan dilakukan oleh panitia yang menghasilkan kunci publik dan kunci privat, masing-masing akan digunakan untuk mengenkripsi dan mendekripsi pilihan suara.

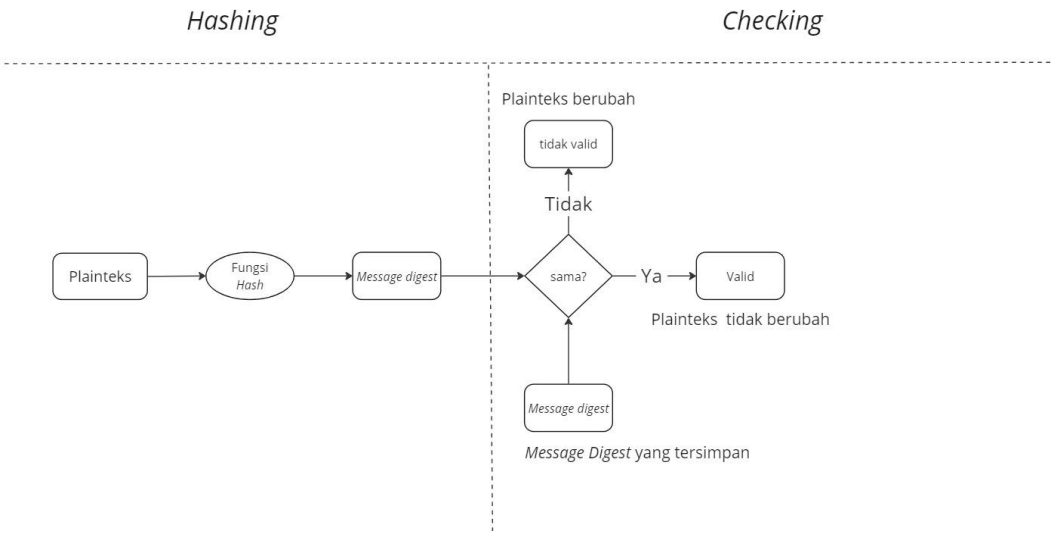
Proses selanjutnya merupakan registrasi pemilih yang memasukan nomor identitas, nama serta nomor TPS yang menghasilkan sebuah token, di mana token ini dihasilkan dari *hashing* menggunakan algoritma SHA-256 dan enkripsi dari kriptografi RSA. Setelah mendapatkan token, pemilih akan memasuki halaman pemungutan suara atau *voting*, setelah melakukan pemilihan akan dihasilkan sebuah cipherteks hasil dari enkripsi token serta pilihan dari pemilih menggunakan kriptografi RSA.

3.2 Model Dasar

Model dasar yang akan digunakan pada penelitian ini merupakan fungsi *hash* serta algoritma kriptografi RSA.

3.2.1 SHA-256

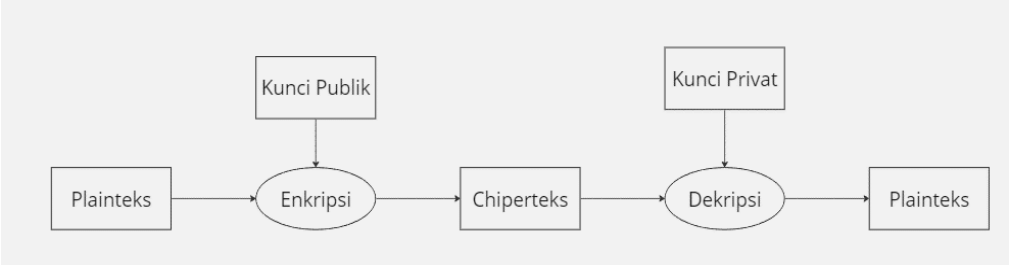
Menurut Stinson (2005) Fungsi *hash* digunakan untuk mengontruksi sebuah "*fingerprint*" yang pendek dari sebuah data. Jika data diubah, maka *fingerprint* tidak akan valid. Bahkan jika sebuah data disimpan di tempat yang tidak aman, itu akan dicek dari waktu ke waktu dan menghitung ulang *fingerprint* dan memverifikasi sebuah *fingerprint* belum berubah. Pada penelitian ini *input* berupa nama lengkap pemilih, nomor TPS serta nomor identitas pemilih yang selanjutnya dilakukan proses *hashing* menggunakan fungsi *hash* SHA-256 untuk mendapatkan *message digest*.



Gambar 3.1 Skema Fungsi Hash

3.2.2 Kriptografi RSA

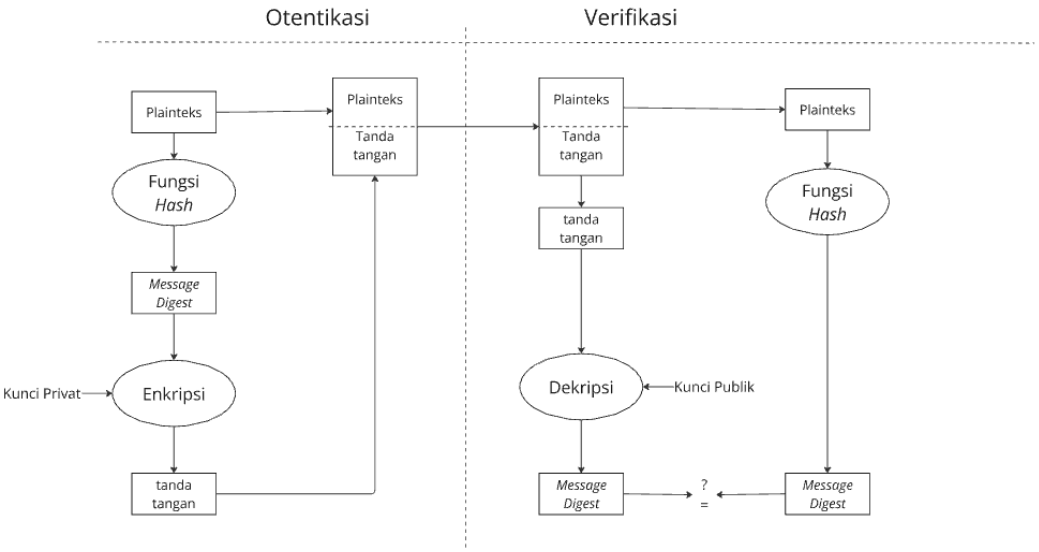
Algoritma RSA ini bekerja menggunakan sepasang kunci, yaitu kunci publik serta kunci privat. Proses enkripsi menggunakan algoritma ini melibatkan kunci publik sedangkan proses dekripsinya menggunakan kunci privat.



Gambar 3.2 Skema Enkripsi dan Dekripsi RSA

3.2.3 Tanda Tangan Digital

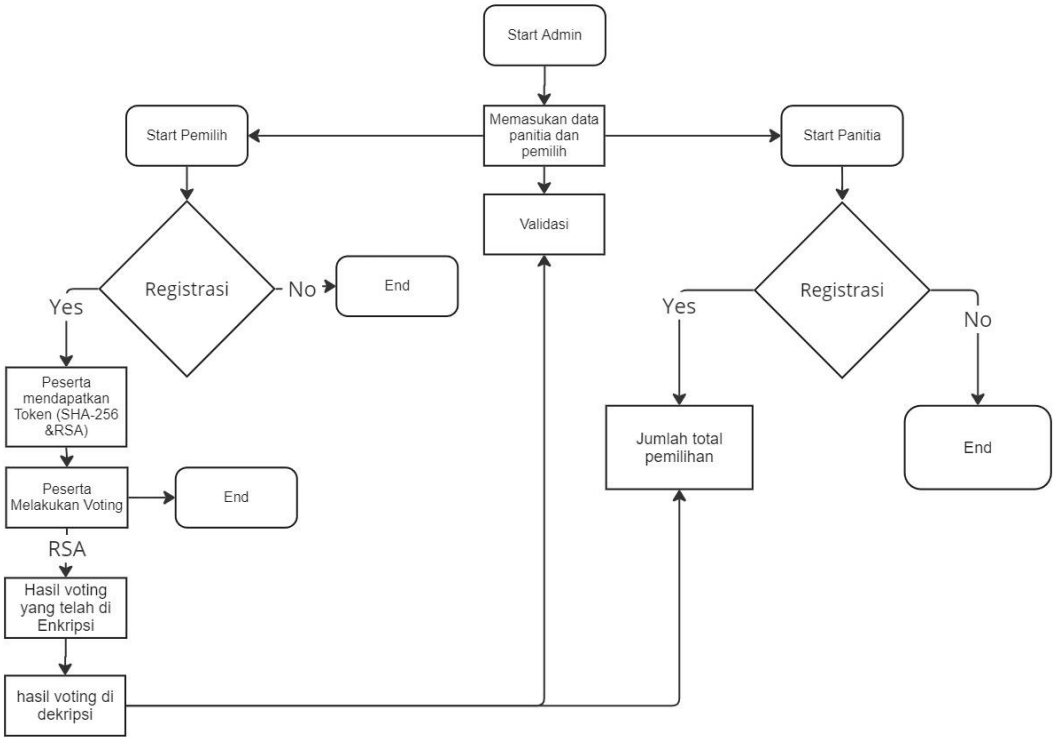
Penandatanganan digital pada penelitian ini menggabungkan antara fungsi *hash* SHA-256 dan algoritma RSA untuk mendapatkan token. Penandatanganan digital digunakan untuk memvalidasi suara yang sah dan tidak pada hasil pemilihan.



Gambar 3.3 Skema Tanda Tangan Digital

3.3 Pengembangan Model

Pengembangan model dasar yang diterapkan dalam penelitian ini adalah gabungan dari skema algoritma RSA serta algoritma SHA-256. Algoritma RSA disini berfungsi untuk merahasiakan pilihan dari pemilih dan algoritma SHA-256 digunakan untuk mengambil nilai identitas pada proses registrasi yang dapat digunakan untuk proses verifikasi.



Gambar 3.4 Skema Pengembangan Model

Berdasarkan Gambar 3.4, pertama admin akan mengunggah data panitia dan pemilih, lalu pemilih akan melakukan registrasi untuk mendapatkan token, token ini didapat dari proses penandatanganan data pemilih menggunakan algoritma SHA-256 dan kriptografi RSA, setelah pemilih melakukan *voting*, pilihan dari pemilih akan dilakukan proses perahasiaan menggunakan algoritma RSA lalu dikirim ke *database*. Setelah pemilihan selesai panitia akan melihat hasil dari pemilihan suara yang sudah diolah oleh sistem.

3.4 Perancangan Program Aplikasi

Program aplikasi akan menggunakan *javascript* dengan rincian sebagai berikut:

3.4.1 Input dan Output

Input untuk program aplikasi ini berupa teks lalu teks ini akan dienkripsi sehingga diperoleh cipherteks. Cipherteks ini akan didekripsi untuk melihat hasil *voting* yang dipilih pemilih.

3.4.2 Protokol

Program aplikasi yang akan dikembangkan akan menggunakan tiga protokol yaitu, protokol untuk admin ,protokol untuk pemilih serta protokol untuk panitia.

a. Protokol Admin

- Admin akan mengunggah data pemilih serta data panitia.
- Admin dapat memvalidasi dari suara pada aplikasi tersebut.

b. Protokol pemilih

- Pemilih memasukan nama lengkap, nomor identitas, serta nomor TPS.
- Pemilih menekan tombol lanjutkan untuk mendapatkan token, di mana token ini merupakan hasil dari *hashing* serta perahasiaan nama lengkap, nomor identitas, serta nomor TPS menggunakan algoritma SHA-256 dan RSA.
- Setelah menekan tombol lanjutkan pemilih akan berada di halaman *voting*.
- Pemilih memilih pilihannya.
- Pemilih menekan tombol selesai, disini sistem akan mengenkripsi token serta pilihan pemilih sebelum dikirimkan ke *database.*

c. Protokol panitia

- Panitia memasukan nama lengkap, serta nomor identitas.
- Jika nomor identitas serta nama lengkap tersebut sesuai yang ada dalam *database* maka panitia akan melanjutkan ke halaman hasil voting.
- Pada halaman hasil voting panitia akan melihat hasil dari keseluruhan voting yang sudah dihitung oleh sistem.

3.4.3 Algoritma Deskriptif

Program aplikasi yang akan dikembangkan akan menggunakan dua algoritma, yaitu *hashsing*, pembangkitan kunci serta enkripsi dan dekripsi kriptografi RSA.

a. Hashing

Hashing dilakukan oleh pemilih dengan memasukan nama, nomor TPS dan nomor identitas pemilih pada aplikasi. Lalu *hashing* dilakukan menggunakan

javascript menggunakan *library* crypto, hasil dari proses ini merupakan sebuah token. Token tersebut digunakan untuk verifikasi pemilih saat sebelum melakukan *voting* lalu token itu juga akan dienkripsi menjadi sebuah tanda tangan digital. Langkah-langkah untuk melakukan *hash* pada program aplikasi sebagai berikut:

1. Pemilih memasukan nama, nomor TPS dan nomor identitas
2. Masukan tersebut akan diproses oleh aplikasi menggunakan *library* crypto
3. Setelah proses selesai maka pemilih akan mendapatkan *output* berupa *Message Digest*.

b. Pembangkitan Kunci

Algoritma pertama ini digunakan oleh panitia untuk menghasilkan sebuah kunci publik yang digunakan untuk autentikasi pemilih dan sebuah kunci privat untuk mendekripsi hasil pemungutan suara.

Proses pembangkitan kunci algoritma RSA sebagai berikut:

1. Panitia memilih dua bilangan prima, p dan q $(p \neq q)$
2. Hitung $n = pq$
3. Hitung $\phi(n) = (p-1)(q-1)$
4. Pilih sebuah bilangan bulat e untuk kunci publik, di mana e relatif prima terhadap $\phi(n)$
5. Hitung kunci dekripsi, d, mengunakan persamaan

$$ed \equiv 1(mod\ \phi(n)) \text{ atau } d \equiv e^{-1} mod(\phi(n))$$

Hasil dari algoritma di atas mendapatkan kunci publik (e, n) dan kunci privat (d, n)

c. Enkripsi dan Dekripsi RSA

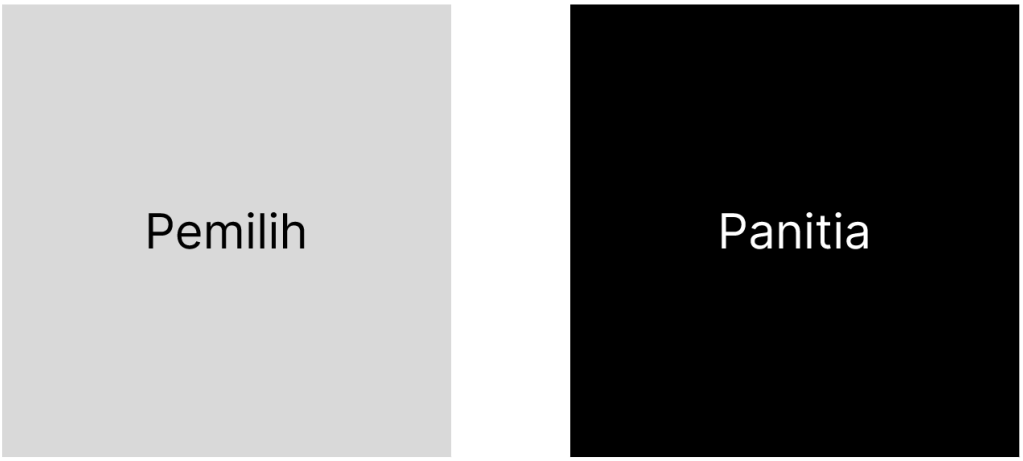
Setelah proses *hashing* dan pembangkitan kunci selesai maka akan dilanjutkan dengan proses enkripsi menggunakan perhitungan yang sudah dijelaskan pada BAB II subbab 2.5. Enkripsi dilakukan untuk mendapatkan cipherteks yang dibutuhkan untuk kerahasiaan pemilih serta kerahasiaan pilihan *voting* pemilih yang dilakukan oleh *user*. Langkah-langkahnya sebagai berikut:

Tabel 3.1 Enkripsi dan Dekripsi RSA

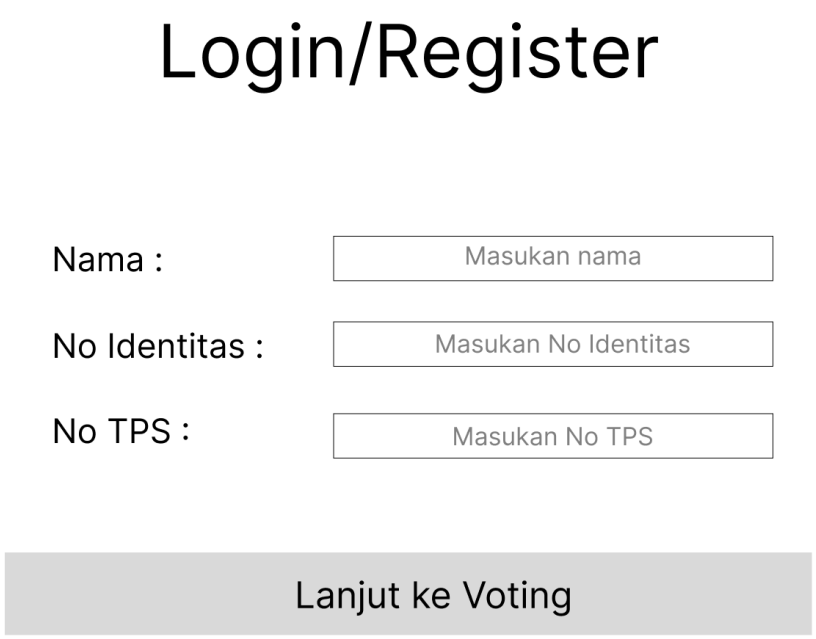
Enkripsi	Dekripsi
• Setelah pemilih melakukan registrasi, pemilih akan mendapatkan token, token ini akan secara otomatis masuk ke dalam *database*. • Pemilih menetukan pilihannya pada program pemungutan suara yang selanjutnya akan dienkripsi bersamaan dengan token yang sudah ada. • Hasil dari enkripsi tersebut akan langsung diunggah ke dalam *database* yang sudah merupakan tanda tangan. • Tanda tangan yang sudah didapat akan ditampikan pada halaman validasi.	• Setelah pemilih melakukan pemilihan. Pilahan, token serta pilihan yang sudah dienkripsi akan langsung didekripsikan. • Setelah hasil dekripsi didapatkan hasil tersebut akan ditampilkan pada halaman validasi. • Admin mencocokkan token dari setiap pemilih dengan token yang didapat dari hasil dekripsi pemungutan suara. • Panitia akan mendapatkan hasil dari pemungutan suara yang merupakan hasil penghitungan dari hasil proses dekripsi.

3.4.4 Rancangan Tampilan Aplikasi Web

Rancangan awal tampilan program aplikasi web yang akan dibuat sebagai berikut:

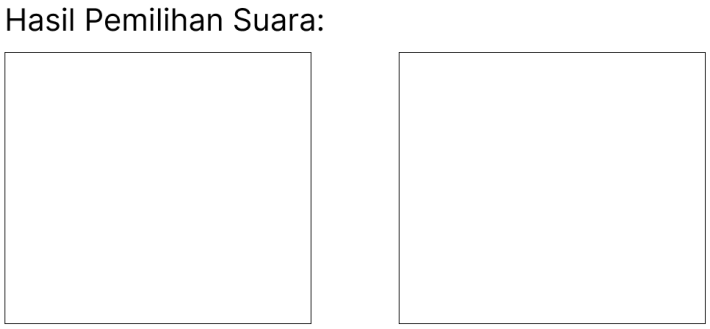


Gambar 3.5 Rancangan Tampilan Halaman Home



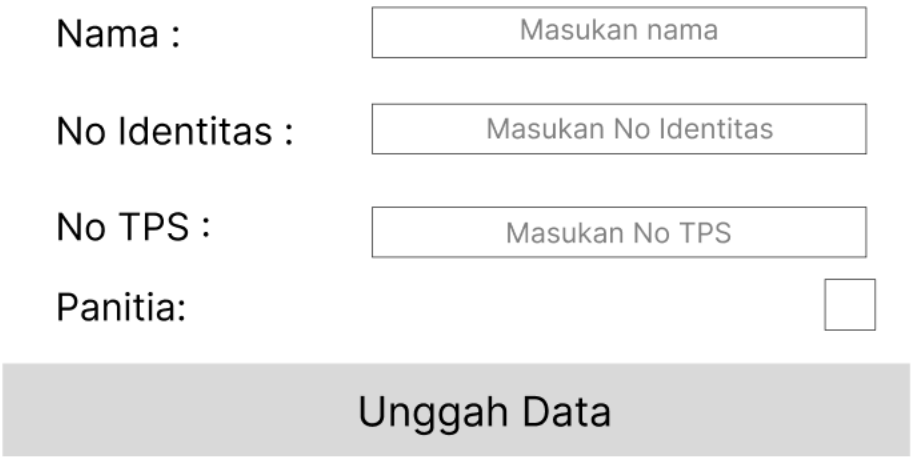
Gambar 3.6 Rancangan Tampilan Halaman Registrasi Pemilih

Gambar 3.7 Rancangan Tampilan Halaman *Voting*



Gambar 3.8 Rancangan Tampilan Halaman Validasi

Gambar 3.9 Rancangan Tampilan Hasil Pemilihan



Gambar 3.10 Rancangan Tampilan Unggah Data Pemilih dan Panitia

Gambar 3.11 Rancangan Tampilan Registrasi Panitia

3.4.5 *Library* Program

Bahasa pemrograman yang dipakai pada penelitian ini adalah *javascript*. Pada penelitian ini ada dua *Framework* utama yang digunakan untuk membangun aplikasi *e-voting* berbasis web yaitu:

- Express.js

  Express.js akan digunakan untuk pembuatan *backend* dari web yang akan dibuat. Pada bagian *backend* ini pada pembuatan *hashing* maupun pada perahasiaan nya akan menggunakan *library* crypto.

- React.js

  React.js akan digunakan untuk pembuatan *frontend* atau tampilan pada web yang akan dibuat.

3.5 Proses Validasi

Dalam tahap validasi ini, admin akan melihat pada halaman validasi. Admin akan membandingkan hasil keluaran dari halaman registrasi atau token dari setiap pemilih dan program pemungutan suara hasil dekripsi dari tanda tangan digital. Jika token dari program registrasi dan token program pemungutan suara cocok pada program validasi pilihan suara, maka suara yang diterima dianggap sebagai suara yang sah.

3.6 Pengambilan Kesimpulan

Pada tahap ini akan ditarik kesimpulan dari keluaran aplikasi web *e-voting* yang telah tervalidasi benar, yaitu terkait dengan penggunaan SHA-256 pada keabsahan suara dan algoritma kriptografi RSA pada perahasiaan pilihan. Setelah tervalidasi, dapat disimpulkan bahwa aplikasi berjalan dengan baik dalam perahasiaan pemilih dan pilihannya.