JTIM : Jurnal Teknologi Informasi dan Multimedia
Vol. 3, No. 4, Pebruari 2022, hlm. 217-226

p-ISSN : 2715-2529
e-ISSN : 2684-9151

ESVISIGN: Tanda Tangan Digital Sekolah Vokasi IPB

(*Esvisign: Digital Signature in College of Vocational School IPB University*)

Walidatush Sholihah[1]*, Sofiyanti Indriasari[2], Inna Novianty[3], Anggi Mardiyono[4], Nur Aziezah[5]

[1],[2],[3],[5]Teknik Komputer, Institut Pertanian Bogor
E-mail: walidah@apps.ipb.ac.id, sofiyanti@apps.ipb.ac.id, innanovianty@apps.ipb.ac.id, nuraziezah@apps.ipb.ac.id
[4]Teknik Informatika dan Komputer, Politeknik Negeri Jakarta
E-mail: anggi.mardiyono@tik.pnj.ac.id

KEYWORDS:
Digital, esvisign, prototype, SHA256, signature

ABSTRACT
All activities have been online since the Covid-19 pandemic broke out in Indonesia. IPB also supports government policy by partially locking down the campus area. Administrative activities are also in online mode. Online correspondence activities still require a signature as a form of validity and authentication of the letter. The scanned signature is different from the digital signature. Thus an application was made to create digital signatures. This application is named eSVi sign. This web-based application was made by using Prototype method. Prototype method consists of communication, planning and modeling quickly, making prototypes, submitting the system to users and feedback. This app uses a hash function with SHA256. Each signed document is assigned a hash value which is stored in storage. When the document is verified, the system will look for a hash value that matches the document in the database. The method to test this application was black box testing. This application can be accessed on the https://ipb.link/esvisign. This digital signature is only used within the College of Vocational Studies IPB University.

KATA KUNCI:
Digital, esvisign, prototipe, SHA256, tanda tangan

ABSTRAK
Semua kegiatan dilaksanakan secara daring sejak pandemi covid-19 mewabah di Indonesia. IPB turut mendukung kebijakan pemerintah dengan kebijakan partially lock down area kampus. Kegiatan administrasi turut dilakukan secara daring. Kegiatan surat menyurat secara daring tetap memerlukan tanda tangan sebagai bentuk kesahan dan validasi dari surat tersebut. Tanda tangan hasil pindai berbeda dengan digital signature. Dengan demikian dibuatlah suatu aplikasi untuk membuat tanda tangan digital. Aplikasi ini diberi nama eSVi sign. Metode pembuatan aplikasi berbasis web ini menggunakan metode Prototipe. Metode Prototipe terdiri atas komunikasi, perencanaan dan pemodelan secara cepat, pembuatan prototipe, penyerahan sistem ke pengguna dan umpan balik. Aplikasi ini menggunakan fungsi hash dengan SHA256. Setiap dokumen yang ditandatangani diberi nilai hash yang disimpan di dalam storage. Ketika dokumen diverifikasi, sistem akan mencari nilai hash yang cocok dengan dokumen tersebut di database. Pengujian yang dilakukan pada aplikasi ini menggunakan metode black box. Aplikasi ini dapat diakses pada laman https://ipb.link/esvisign. Tanda tangan digital ini hanya digunakan di lingkup Sekolah Vokasi IPB.

I. PENDAHULUAN

Penyakit covid-19 melanda Indonesia sejak Maret 2020. Terkait hal ini, untuk mengurangi jumlah penderita, pemerintah telah menetapkan berbagai kebijakan diantaranya bekerja dan belajar dari rumah. IPB University turut mendukung kebijakan pemerintah dengan pembatasan masuk kampus. Semua pegawai dan dosen bekerja dari rumah. Layanan kemahasiswaan sampai ujian dilakukan secara daring. Walaupun saat ini sudah dilakukan relaksasi dan "normal baru", namun beberapa hal termasuk kegiatan administrasi masih dilakukan secara daring.

Kegiatan administrasi seperti persuratan menggunakan tanda tangan dan stempel. Berdasarkan Kamus Besar Bahasa Indonesia, tanda tangan sebagai lambang nama yang dituliskan

* Penulis Korespondensi (Walidatush Sholihah)
Email : walidah@apps.ipb.ac.id

dengan tangan oleh orang itu sendiri sebagai penanda pribadi (telah menerima dan sebagainya). Tanda tangan diberikan pada suatu dokumen untuk menandakan legalitas dan verifikasi dokumen tersebut [1], [2]. Agar surat atau dokumen elektronik tetap dapat diakui keabsahannya, maka digunakanlah *digital signature.*

Digital signature sering digunakan dalam berbagai transaksi elektronik. Berdasarkan Undang-Undang Nomor 11 Tahun 2008 tentang Informasi dan Transaksi Elektronik (UU ITE), transaksi elektronik adalah perbuatan hukum yang dilakukan dengan menggunakan komputer, jaringan komputer, dan/atau media elektronik lainnya. Transaksi elektronik seringkali disebut *electronic commerce* atau e-commerce lebih banyak memanfaatkan teknologi internet. Internet merupakan kumpulan jaringan di seluruh dunia yang menghubungkan jutaan bisnis, pemerintahan, institusi pendidikan, dan individu [3]. Selain itu digital signature juga digunakan pada *electronic voting* [4] , lelang [5], rumah sakit [6], [7], surat kuasa [8].

Tanda tangan hasil pindai berbeda dengan tanda tangan digital (*digital signature*). Tanda tangan digital atau tanda tangan elektronik berdasarkan UU ITE pasal 1 ayat 12 adalah tanda tangan yang terdiri atas Informasi Elektronik yang dilekatkan, terasosiasi atau terkait dengan Informasi Elektronik lainnya yang digunakan sebagai alat verifikasi dan autentikasi. UU ITE pasal 1 ayat 1 menyatakan informasi elektronik adalah satu atau sekumpulan data elektronik, tetapi tidak terbatas pada tulisan, suara, gambar, peta, rancangan, foto, *electronic data interchange* (EDI), surat elektronik (electronic mail), telegram, teleks, telecopy atau sejenisnya, huruf, tanda, angka, Kode Akses, simbol, atau perforasi yang telah diolah yang memiliki arti atau dapat dipahami oleh orang yang mampu memahaminya. Tanda tangan elektronik ada yang tersertifikasi dan tidak tersertifikasi berdasarkan Peraturan Pemerintah RI tahun 2012 pasal 54. Tanda tangan elektronik yang tersertifikasi harus dibuat dengan menggunakan jasa penyelenggara sertifikasi elektronik dan dibuktikan dengan sertifikat elektronik. Sedangkan tanda tangan elektronik tidak tersertifikasi dibuat tanpa menggunakan jasa penyelenggara sertifikasi elektronik.

Penggunaan tanda tangan digital sudah merambah berbagai bidang. Peraturan tentang tanda tangan digital di dunia dikaji terkait dampaknya terhadap perkembangan perdagangan internasional [9]. Banyak negara sudah mengaplikasikan *digital signature*, diantaranya Malaysia [10], Austria [11]. Inggris [12]. Walaupun dinyatakan aman, penggunaan tanda tangan elektronik tetap memiliki resiko keamanan dan dapat disalahgunakan terutama ketika terhubung dengan internet [13].

Penelitian-penelitian mengenai digital signature sudah banyak dilakukan. Pada tahun 2008, Liu dan Li mengemukakan skema tanda tangan digital tanpa fungsi hash satu arah. Tetapi skema Liu dan Li ini ternyata tidak aman [14]. Skema tanda tangan *pairing-free certificateless* menggunakan *elliptic curve cryptography* [15] juga dibuktikan kurang aman pada beberapa teknik pemalsuan [16]. Teknik *Batch Verification* memverifikasi beberapa tanda tangan digital sekaligus [17]. Teknik ini memiliki beberapa variasi dan beban komputasinya pun rendah. Penelitian tentang *storage* untuk menyimpan kunci tanda tangan digital sudah dilakukan dengan berbagai metode [18], [19]. Algoritma RSA tanda tangan digital yang efisien dilakukan dengan mengurangi jumlah bilangan prima yang diperlukan [20]. Adanya komputer kuantum menjadi ancaman bagi tanda tangan digital karena komputer kuantum dapat menghitung bilangan yang sangat besar. Salah satu skema yang tahan terhadap serangan komputer kuantum yaitu *Smart Digital Signature* (SDS) [21]. Penelitian lain mengenai perlindungan dokumen yaitu dengan metode pengenalan biometrik. Metode ini menggunakan gambar sinyal ucapan dari sonogram [22]. Selain tanda tangan digital, autentikasi dokumen juga bisa dilakukan dengan menempelkan suatu *watermark* ke dokumen sebagai bagian dari dokumen itu sendiri [23]. JadES signatures memungkinkan validasi tanda tangan yang sudah habis masa sertifikasinya [24]. Skema tanda tangan digital yang berdasarkan *expanded root problem* memungkinkan untuk membangun platform tanda tangan digital dengan keamanan tinggi [25].

Selama pandemi, kegiatan surat menyurat di Sekolah Vokasi IPB (SV IPB) menggunakan tanda tangan dan stempel hasil pindai (*scan*). Tanda tangan dan stempel hasil pindai sangat rentan untuk dipalsukan dan sulit diverifikasi keasliannya. Terkait dengan hal tersebut, setiap surat yang dibuat oleh Sekolah Vokasi IPB tentunya harus dapat diverifikasi keasliannya. Keaslian yang dimaksud disini adalah keaslian tanda tangan yang tercantum

dan surat tersebut benar-benar dikeluarkan oleh SV IPB. Bagian yang sering membuat surat di SV IPB yaitu bagian akademik, bagian komisi disiplin dan kemahasiswaan, bagian komisi konseling, dan organisasi kemahasiswaan.

Berdasarkan paparan di atas dapat disimpulkan bahwa kegiatan administrasi berupa surat menyurat dapat dibuat dalam bentuk elektronik dengan memanfaatkan digital signature. Surat tersebut harus dapat dilihat dan dibaca oleh siapa saja. Tetapi isi surat harus tetap asli (tidak berubah). Tanda tangan pada surat harus dapat diyakinkan milik orang yang bersangkutan.

Tujuan yang hendak dicapai dalam penelitian ini adalah membangun aplikasi berbasis web ”eSVi sign” untuk kegiatan tanda tangan dan verifikasi surat elektronik di lingkungan Sekolah Vokasi IPB. Aplikasi eSVi-sign menyediakan fasilitas untuk membuat tanda tangan digital pada surat, menyediakan fasilitas verifikasi surat yang sudah ditandatangani dan otomatisasi kegiatan surat menyurat sehingga proses menjadi *paperless*. Tanda tangan digital yang dibuat pada penelitian ini merupakan tanda tangan digital tidak tersertifikasi dan hanya berlaku di lingkungan Sekolah Vokasi IPB.

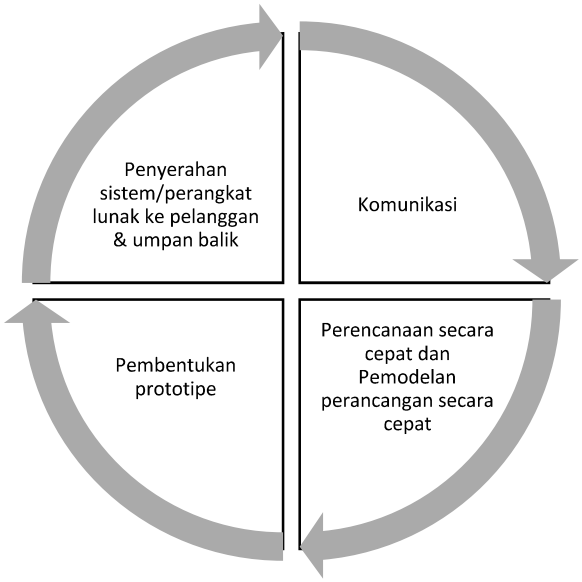
II. METODOOGI

Metode penelitian yg digunakan pada eSVi-sign: tanda tangan digital Sekolah Vokasi IPB ini menggunakan metode pengembangan perangkat lunak model prototipe [26]. Metode prototipe terdiri atas lima tahap yaitu komunikasi; perencanaan secara cepat; pemodelan perancangan secara cepat; pembentukan prototipe; dan penyerahan sistem/perangkat lunak ke pelanggan serta umpan balik. Tahapan dalam metode prototipe dapat dilihat pada Gbr. 1. Tahapan-tahapan pada metode prototipe sebagai berikut:

A. Komunikasi

Tahap awal yang dilakukan dalam metode prototipe yaitu komunikasi antara pihak pengembang dengan pihak pelanggan. Proses yang dilakukan pada tahap ini adalah mendefinisikan kebutuhan apa saja untuk perangkat lunak yang akan dikembangkan dan mengidentifikasi spesifikasi kebutuhan apapun yang saat ini diketahui.

Ada beragam metode yang digunakan untuk berkomunikasi dengan pihak pelanggan salah satunya yaitu dapat dilakukan dengan cara diskusi dan wawancara. Tujuan dari diskusi dan wawancara itu sendiri untuk mengajukan pertanyaan-pertanyaan yang berhubungan dengan aktivitas saat ini, menemukan fakta, mendapatkan data yang akurat, mengumpulkan ide dan pendapat dengan pihak pelanggan yang menginginkan dibangunnya sebuah sistem. Diskusi dan wawancara memiliki keunggulan dibanding metode lain karena dapat mengetahui hal-hal detail yang mungkin sulit diungkap sebelumnya. Selain itu, tim dan pewawancara juga akan mendapatkan jawabannya saat itu juga [27].



Gbr. 1 Metode Prototipe

B. Perencanaan secara cepat dan pemodelan perancangan secara cepat

Iterasi pembuatan prototipe direncanakan dengan cepat dan pemodelan dalam bentuk rancangan cepat dilakukan. Rancangan cepat akan memulai konstruksi pembuatan suatu prototipe [26]. Setelah perencanaan, langkah selanjutnya adalah pembuatan model sistem. Tujuan dari model sistem adalah untuk menyatukan pemahaman tentang sistem yang dihasilkan dan dengan cepat menganalisa siapa saja aktor yang terlibat di dalam sistem. Pengguna dapat melihat pemodelan dalam bentuk *use case* diagaram, activity diagram dan *class diagram*.

Use case merupakan pemodelan kelakuan (*behavior*). Diagram *use case* akan menggambarkan apa yang dikerjakan oleh masing-masing aktor. *Use case* juga digunakan untuk mengetahui fungsi apa saja yang ada di dalam sebuah sistem informasi dan

siapa saja yang berhak menggunakan fungsi-fungsi itu [28].

C. Pembentukan Prototipe

Tahap pembentukan prototipe merupakan tahap untuk menerapkan tahapan perancangan. Pembuatan dan pengkodean perangkat lunak meliputi tahap implementasi database, tahap implementasi antarmuka, tahap implementasi masukan, tahap implementasi proses, dan tahap implementasi keluaran.

D. Penyerahan sistem/perangkat lunak ke pelanggan serta umpan balik

Prototipe kemudian diserahkan kepada pelanggan untuk dilakukan evaluasi terhadap aplikasi. Pelanggan akan mengevaluasi dan memberikan umpan balik terhadap aplikasi apakah sudah memenuhi kebutuhan atau belum. Jika kebutuhan belum terpenuhi, maka prototipe akan diperbaiki dan dilakukan iterasi kedua sampai semua kebutuhan pengguna terpenuhi.

Setelah melakukan implementasi tahap selanjutnya adalah pengujian. Pengujian dilakukan dengan menggunakan metode black-box testing. Pengujian dilakukan untuk menemukan kesalahan pada aplikasi dan mendemonstrasikan fungsional aplikasi saat dioperasikan.

III. HASIL DAN PEMBAHASAN

Kegiatan penelitian ini mengikuti metode Prototipe [26]. Hasil dan pembahasan dari penelitian ini dipaparkan sebagai berikut:

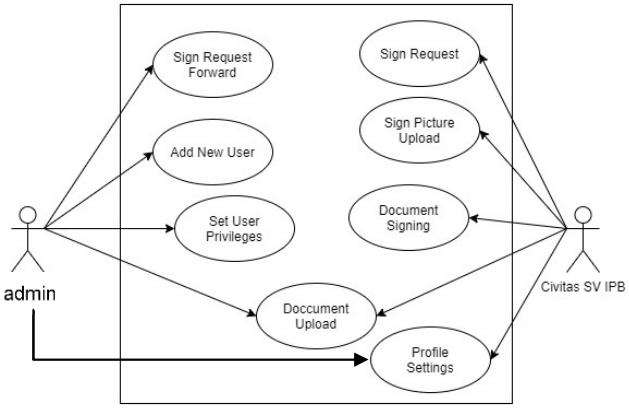
A. Komunikasi

Pada tahap ini dilakukan pembicaraan dan penyamaan persepsi tentang aplikasi yang akan dibuat. Penyamaan persepsi dilakukan antara peneliti dengan tim programmer. Hasil yang diperoleh berupa batasan dan fungsionalitas dari aplikasi eSVi Sign. Fungsionalitas aplikasi eSVi Sign sebagai berikut:

1. Pengguna dapat menandatangani dokumen secara digital.
2. Pengguna dapat mengunggah gambar untuk tanda tangan.
3. Pengguna dapat mengunduh dokumen yang sudah ditandatangani.
4. Pengguna dapat mengajukan permohonan tanda tangan kepada lebih dari satu orang.
5. Pengiriman notifikasi atau dokumen melalui email.
6. Verifikasi tanda tangan dapat dilakukan di aplikasi eSVi-Sign dan aplikasi pdf viewer.
7. Aplikasi eSVi-Sign menggunakan pustaka Signer untuk memberikan tanda tangan digital pada berkas dengan keluaran berupa file pdf.

B. Perencanaan secara cepat dan pemodelan perancangan secara cepat

Pada tahap perencanaan secara cepat, dilakukan perancangan use case dan class diagram dari aplikasi yang akan dibangun. Use case diagram menyatakan visualisasi interaksi yang terjadi antara pengguna (aktor) dengan sistem [29]. *Use case* diagram pada aplikasi eSVi Sign disajikan pada Gbr. 2. Pada aplikasi ini terdapat dua aktor yang berinteraksi dengan sistem. Aktor yang berinteraksi adalah admin dan civitas SV IPB. Hak akses yang dimiliki admin dan civitas yaitu menandatangani dokumen secara digital, meminta (*request*) tanda tangan digital, mengunggah dokumen, dan mengatur (*setting*) profil. Hak akses yang hanya dimiliki oleh administrator dan tidak dimiliki aktor lain yaitu menambah pengguna dan menentukan hak akses pengguna baru tersebut.



Gbr. 2. Diagram *Use Case*

Algoritme penandatanganan terdiri dari tiga bagian yaitu pembangkitan pasangan kunci, pembangkitan tanda tangan dan verifikasi. Algoritma *digital signature* [30] sebagai berikut:

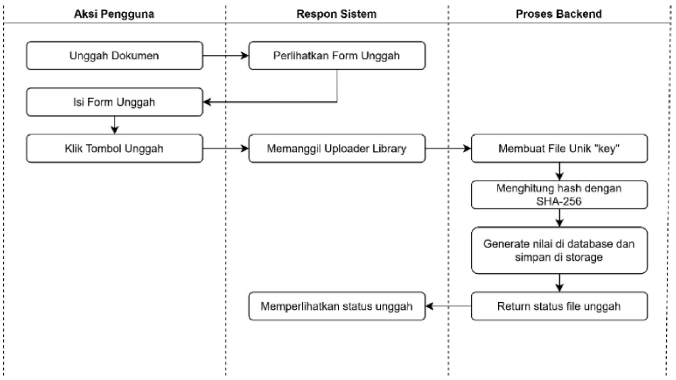
1. Pembangkitan pasangan kunci
   a. Pilih bilangan prima q sehingga $2^{159} < q < 2^{160}$.
   b. Pilih t $(0 \leq t \leq 8)$ dan pilih bilangan prima p dimana $2^{511+64t} < p < 2^{512+64t}$, dengan q membagi (p-1).

c. Pilih generator α anggota grup siklik unik berorder q di dalam $\mathbb{Z}_p^*$
   (i) Pilih elemen $g \in \mathbb{Z}_p^*$ dan hitung $\alpha = g^{(p-1)/q} \, mod \, p$.
   (ii) Jika $\alpha = 1$, kembali ke langkah (i).
d. Pilih bilangan bulat secara acak x sehingga $1 \leq x \leq q-1$.
e. Hitung $y = \alpha^x \, mod \, p$
f. Hasil: kunci publik : (p, q, α, y) dan kunci privat : x.

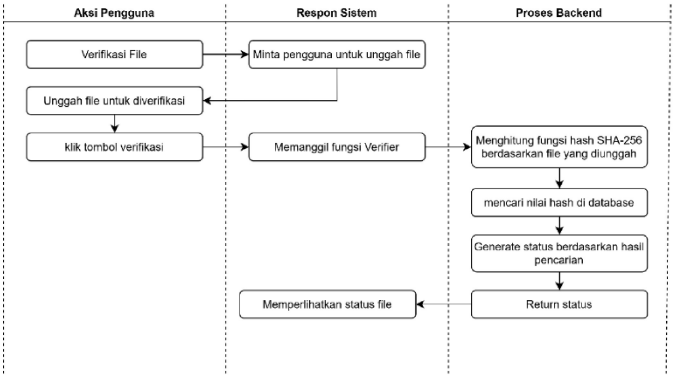
2. Pembangkitan tanda tangan
   a. Tentukan bilangan bulat acak secara rahasia k, $0 < k < q$.
   b. Hitung $r = (\alpha^k \, mod \, p) \, mod \, q$.
   c. Hitung $k^{-1} \, mod \, q$.
   d. Hitung $s = k^{-1}\{h(m) + xr\} \, mod \, q$. $h(m)$ adalah fungsi hash dari pesan m.
   e. Tanda tangan untuk pesan m adalah pasangan (r, s).
3. Verifikasi tanda tangan
   a. Pihak yang akan memverifikasi tanda tangan memiliki kunci publik (p, q, α, y).
   b. Periksa $0 < r < q$ dan $0 < s < q$. Jika tidak, maka tolak tanda tangan.
   c. Hitung $w = s^{-1} \, mod \, q$.
   d. Hitung $u_1 = w.h(m) \, mod \, q$ dan $u_2 = rw \, mod \, q$.
   e. Hitung $v = (\alpha^{u_1} y^{u_2} \, mod \, p) \, mod \, q$.
   f. Terima tanda tangan jika dan hanya jika $v = r$.

Proses yang dilakukan pada aplikasi eSVi Sign secara umum terdiri atas tiga proses. Ketiga proses tersebut yaitu unggah dokumen, verifikasi dokumen dan penandatanganan dokumen. Proses unggah dokumen (Gbr. 3) dimulai dari saat pengguna mengunggah dokumen ke aplikasi eSVi Sign. Pengguna mengisi form unggah. Setelah itu, sistem memanggil *library uploader* dan membuat file unik. File unik inilah yang disebut sebagai kunci privat. Kunci ini dihasilkan dari fungsi hash. Fungsi hash yang digunakan yaitu SHA256. Algoritma SHA256 menghasilkan *message digest* sebesar 256 bit [31]. Nilai hash ini kemudian disimpan di *storage*.

Proses berikutnya yaitu verifikasi dokumen (Gbr. 4). Dokumen yang akan diverifikasi harus diunggah ke dalam sistem oleh pengguna. Sistem melakukan verifikasi dengan cara memanggil fungsi *verifier*. Fungsi ini akan menghitung nilai hash SHA256 dari file yang diunggah. Kemudian, nilai hash akan dicari di database. Jika menemukan kesamaan, maka dokumen tersebut valid. Jika tidak ada nilai hash yang sama dengan database, maka dokumen tidak dapat diverifikasi oleh sistem.



Gbr. 3. Proses Unggah Dokumen



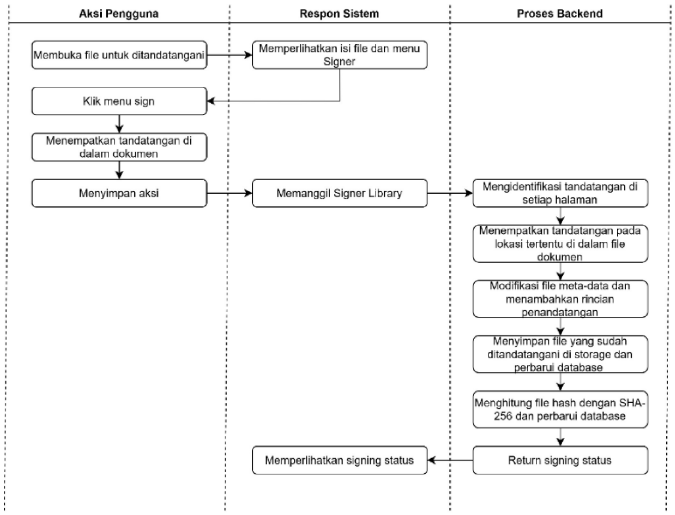
Gbr. 4. Proses Verifikasi Dokumen

Proses ketiga yaitu proses penandatanganan dokumen (Gbr. 5). Dokumen yang akan ditandatangani diunggah terlebih dahulu ke dalam sistem. Pada proses ini, pengguna dapat menempatkan "gambar/ilustrasi tandatangannya" di mana saja di dalam dokumen. Sistem memanggil *library signer*. *Signer* ini akan mengidentifikasi gambar tanda tangan yang ada di setiap halaman pada dokumen. Kemudian, sistem akan menambahkan rincian penandatangan dan menyimpan file yang sudah ditandatangani ke dalam storage.

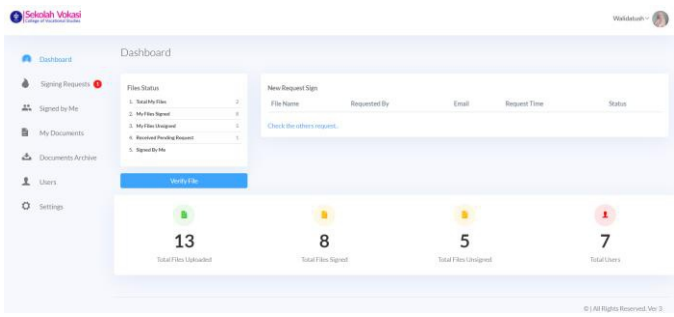
C. Pembentukan Prototipe

Pada tahap pembentukan prototipe, tampilan aplikasi eSVi Sign sudah berhasil dibuat. Ketika aplikasi diakses, pengguna akan diarahkan ke halaman login. Hanya pengguna yang sudah terdaftar saja yang dapat masuk ke dalam aplikasi. Pengguna dapat didaftarkan oleh administrator ke dalam sistem. Setelah login, pengguna akan masuk

ke halaman *dashboard* (Gbr. 6). Di halaman *dashboard*, terdapat beberapa menu yang dapat diakses. Menu yang terdapat pada aplikasi ini yaitu: *Documents, Signed by me, Signing request, Document Archive, Users*, dan *Setting*. Halaman *dashboard* pada Gbr. 6 merupakan halaman dashboard halaman administrator. Administrator dapat melihat jumlah pengguna yang terdaftar di aplikasi eSVi Sign. Di halaman *dashboard* bagian kanan, terdapat beberapa informasi seperti total my files yang memperlihatkan jumlah file yang dimiliki di dalam sistem, jumlah file yang diunggah, jumlah pengguna, permintaan tandatangan (*pending* dan *received*).



Gbr 5. Proses Penandatanganan Dokumen

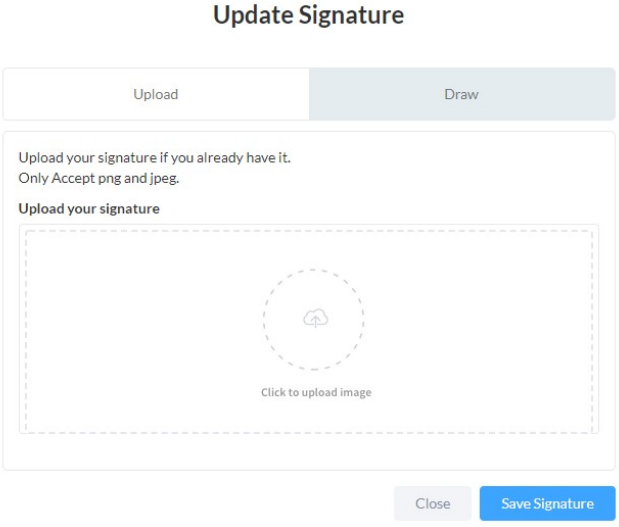


Gbr 6. Halaman *Dashboard*

Menu *Documents* berisi dokumen-dokumen yang sudah diunggah ke dalam sistem. Dokumen yang sudah ditandatangani dan belum, dapat dilihat di menu ini. Semua dokumen dapat dirapikan dan dimasukkan ke dalam folder. Hal ini akan memudahkan pengguna dalam melakukan pencarian dan pengarsipan dokumen. Menu *document archive* memperlihatkan dokumen-dokumen apa saja yang sudah diunggah ke dalam sistem beserta rincian dokumen tersebut. Selain itu, diperlihatkan pula status dokumen, apakah sudah ditandatangani atau belum. Halaman menu *Signed by Me* berisi dokumen-dokumen yang sudah ditandatangani oleh pengguna. Jika dokumen akan diunduh, pengguna dapat klik kanan pada dokumen dimaksud lalu memilih tombol *download*. Halaman *Signing request* memperlihatkan permintaan tanda tangan yang ditujukan ke pengguna. Pengguna dapat menandatangani atau menolak untuk menandatangani dokumen. Jika dokumen sudah ditandatangani, maka orang yang meminta tanda tangan dan yang memberi tanda tangan akan memiliki file yang sama.

Menu user hanya dimiliki oleh pengguna dengan hak akses administrator. Menu ini memperlihatkan daftar pengguna yang ada di dalam sistem. Administrator dapat menambah dan menghapus pengguna melalui menu ini.

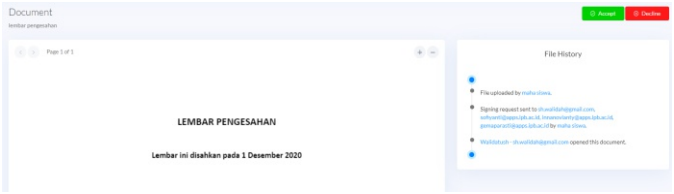
Menu *setting* berisi dua submenu yaitu *profile* dan *signature*. Menu *profile* berisi identitas pengguna. Pada menu *signature*, pengguna dapat memasukkan gambar tanda tangan atau gambar lain sebagai identitas pengguna. *Signature* ini hanya dapat berisi satu gambar dan dapat diperbarui. Jika pengguna ingin memperbarui gambar tanda tangannya, pengguna dapat melakukan *update signature* dan mengunggah gambar atau menggambar tanda tangan barunya (Gbr. 7).



Gbr 7. Menu *Update Signature*

Proses atau kegiatan penandatanganan suatu dokumen pada aplikasi eSVi Sign dapat dijelaskan sebagai berikut:

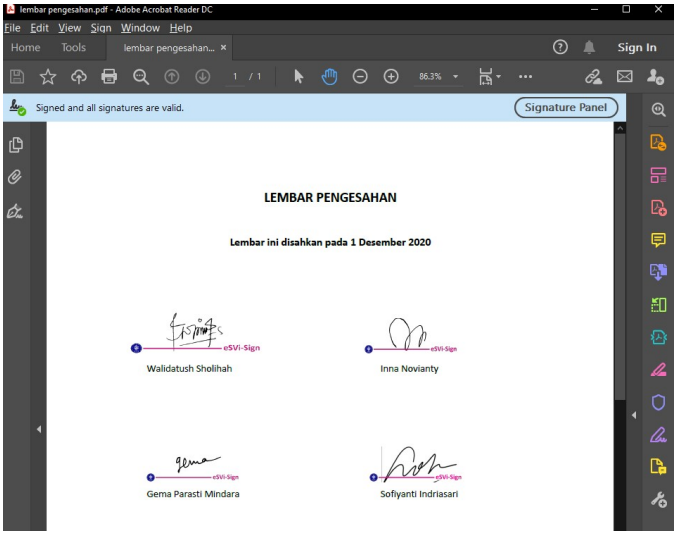
1. Pengguna masuk ke aplikasi eSVi sign dengan memasukkan username dan password masing-masing.
2. Pengguna masuk ke menu documents. Setelah itu pengguna mengunggah dokumen yang akan ditandatangani ke eSVi sign.
3. Pengguna dapat menandatangani dokumen tersebut di menu Documents dengan cara klik kanan file dokumennya lalu memilih open untuk membuka dokumen atau bisa juga dilakukan dengan melaukan klik dua kali pada file tersebut.
4. Jika pengguna akan menandatangani sendiri dokumen tersebut, maka pengguna dapat langsung memilih Sign & Edit. Tetapi jika akan meminta tanda tangan dari pengguna lain, maka pengguna dapat memilih Request Sign.
5. Dokumen yang akan ditandatangani oleh pengguna lain, harus dituliskan dulu siapa saja pengguna yang akan menandatanganinya. Tidak ada urutan penandatanganan pada sistem ini.
6. Pengguna yang mendapat permintaan tanda tangan akan menerima notifikasi berupa email. Pengguna lalu menandatanganinya dengan membuka aplikasi eSVi sign ke menu Signing Request. Di jendela sebelah kanan saat membuka Signing Request akan terlihat file history. Untuk menanda tangani dokumen, pengguna memilih tombol Accept. Setelah itu, pengguna mengatur lokasi untuk menempatkan tanda tangan pada dokumen.
7. Dokumen yang sudah ditanda tangani, disimpan oleh sistem ke dalam storage. Pengguna dapat mengetahui apakah dokumen sudah lengkap ditandatangani dengan melihat di menu signing request pada tab sent. Selain itu, saat dokumen dibuka di menu Documents, juga akan terlihat riwayat penandatanganan (Gbr. 8).
8. Jika dokumen sudah lengkap ditandatangani, maka pengguna dapat mengunduh dokumen tersebut di menu Documents.



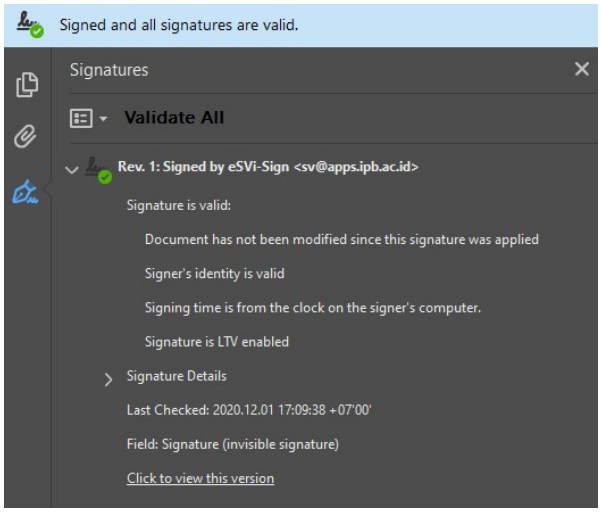
Gbr 8. Menu Signing Request

Dokumen atau file yang sudah ditandatangani melalui aplikasi eSVi Sign tidak terlihat berbeda jika dibandingkan dengan dokumen lainnya secara kasat mata. Namun dokumen tersebut telah diberikan sertifikat digital. Dokumen dapat diverifikasi menggunakan aplikasi pdf reader secara offline atau pun diverifikasi melalui aplikasi eSVi sign secara online. Untuk memverifikasi dokumen secara online melalui aplikasi eSVi sign, pengguna dapat membuka aplikasi eSVi sign melalui browser tanpa harus login, kemudian memasukkan file yang akan diverifikasi. Sistem akan memeriksa file dengan file yang ada di database. Jika file ditemukan, maka file tersebut valid.

Proses verifikasi dokumen secara offline dapat dilakukan dengan aplikasi pdf reader yang tersedia. Dalam hal ini, dokumen dibuka menggunakan aplikasi Adobe Acrobat Reader. Gbr. 9 memperlihatkan tampilan saat file dibuka menggunakan Adobe Acrobat Reader. Secara otomatis, aplikasi tersebut memunculkan *Signature Panel* dan terdapat notifikasi "*signed and all signature are valid*". Jika *Signature Panel* dibuka, terlihat rincian dari dokumen bahwa dokumen tersebut belum pernah dimodifikasi sejak ditandatangani (Gbr. 10).



Gbr 9. Verifikasi *Offline* Menggunakan Adobe Acrobat Reader



Gbr 10. Isi Signature Panel

D. Penyerahan sistem/perangkat lunak ke para pelanggan/pengguna, pengiriman, dan umpan balik

Aplikasi eSVi Sign merupakan aplikasi berbentuk web. Aplikasi ini dapat diakses dimana saja dan kapan saja selama pengguna terhubung dengan internet. Aplikasi eSVi sign juga dapat dibuka di komputer maupun komputer genggam melalui browser. Aplikasi eSVi sign dapat diakses di laman https://ipb.link/esvisign atau di https://klik.sv.ipb.ac.id/svsign.

Aplikasi eSVi sign diuji dengan metode black box. Pengujian black box diarahkan kepada fungsional dari sistem dengan cara memasukkan nilai input yang benar dan nilai input yang salah beserta hasil yang diharapkan. Beberapa pengujian black box aplikasi eSVi sign disajikan pada Tabel I sampai Tabel VI.

TABEL I
PENGUJIAN LOGIN

Nilai *Input*	Skenario Pengujian	Hasil yang diharapkan	Status
Benar	Mengisi data pada *field 'email address'* dan *'password'* lalu menekan tombol *'sign in'*	Menampilkan halaman dashboard dan pesan "Disclaimer, Tanda tangan digital ini hanya berlaku di lingkungan Sekolah Vokasi IPB"	Sukses
Salah	Mengosongkan *field 'email address'* dan *'password'*, lalu menekan tombol *'sign in'*	Menampilkan pesan *'this value is required'*	Sukses

TABEL II
PENGUJIAN *VERIFY FILE*

Nilai *Input*	Skenario Pengujian	Hasil yang diharapkan	Status
Benar	Mengunggah file dengan menekan tombol *choose file*	Membuka halaman untuk unggah file	Sukses
Benar	File yang dipilih akan diverifikasi dengan menekan tombol *'verify'*	Menampilkan notifikasi identitas *signer*	Sukses
Benar	Membatalkan *verify file*	Kembali ke halaman sebelumnya	Sukses
Salah	Mengunggah file yang tidak terdaftar dalam *database,* lalu menekan tombol *'verify'*	Menampilkan pesan "*Not Found. Cannot validate file. This document has been modified or not created by our system*"	Sukses
Salah	Tidak memilih file untuk diunggah, lalu menekan tombol *'verify'*	Menampilkan pesan *'No file chosen'* dan *'this value is required'*	Sukses

TABEL III
PENGUJIAN UBAH *PROFILE*

Nilai *Input*	Skenario Pengujian	Hasil yang diharapkan	Status
Benar	Mengisi data pada *field-field* yang berada di halaman profile, lalu menekan tombol *'save changes'*	Data tersimpan, dan menampilkan pesan *'Alright, profile successfully updated'*	Sukses

TABEL IV
PENGUJIAN *UPDATE SIGNATURE*

Nilai *Input*	Skenario Pengujian	Hasil yang diharapkan	Status
Benar	Menekan tombol *Signature* di menu Setting	Menampilkan halaman *Signature*	Sukses
Benar	Mengubah tanda tangan dengan menekan tombol *Update Signature*	Tanda tangan dapat diperbaharui	Sukses

TABEL V
PENGUJIAN *CREATE FOLDER*

Nilai *Input*	Skenario Pengujian	Hasil yang diharapkan	Status
Benar	Mengisi nama folder pada *field 'folder name'*, lalu menekan tombol *create folder*	Menampilkan folder baru	Sukses
Salah	Mengosongkan *field 'folder name'*, lalu menekan tombol *create folder*	Menampilkan pesan *'this value is required'*	Sukses
Benar	Menutup kolom *create folder* dengan menekan tombol close	Kolom tertutup dan menampilkan halaman sebelumnya	Sukses

TABEL VI
PENGUJIAN *REQUEST SIGN*

Nilai *Input*	Skenario Pengujian	Hasil yang diharapkan	Status
Benar	Tekan tombol '*Request Sign*' yang terdapat pada halaman dokumen	Menampilkan form *'Signature Request'*	Sukses
Benar	Mengisi *field-field* pada form *'Signature Request'* lalu menekan tombol *Send Request*		Sukses

Nilai *Input*	Skenario Pengujian	Hasil yang diharapkan	Status
Salah	Tidak mengisi field-field pada form *'Signature Request'* lalu menekan tombol *Send Request*	Menampilkan pesan "this value is required"	Sukses
Benar	Menekan tombol *close*	Menutup form Signature Request	Sukses

IV. KESIMPULAN

Aplikasi eSVi sign telah berhasil dibuat. Aplikasi ini dapat digunakan untuk menandatangani dokumen elektronik secara digital. Dokumen yang sudah ditandatangani dapat diverifikasi secara *online* melalui aplikasi eSVi sign maupun secara *offline* menggunakan aplikasi pdf reader. Lingkup penggunaan aplikasi ini yaitu sivitas akademika Sekolah Vokasi IPB. Aplikasi eSVi sign dapat diakses secara online pada laman https://ipb.link/esvisign atau https://klik.sv.ipb.ac.id/svsign. Aplikasi dapat diakses melalui *browser* dengan komputer atau komputer genggam yang terhubung dengan internet.

Saat ini aplikasi eSVi sign hanya dapat melakukan tanda tangan pada setiap halaman dokumen. Aplikasi belum dapat melakukan penandatanganan pada banyak dokumen sekaligus atau pada banyak halaman sekaligus. Pada penelitian selanjutnya, dapat ditambahkan fitur *bulk signature* agar aplikasi dapat menandatangani dokumen secara massal sekaligus.

Pengujian kekuatan fungsi hash belum dilakukan pada penelitian ini. Karena pada penelitian ini, aplikasi eSVi sign, menggunakan pustaka (library) signer yang sudah jadi (*fixed library*). Pada penelitian selanjutnya dapat diuji kekuatan dari fungsi hash yang digunakan.

UCAPAN TERIMA KASIH

Terima kasih penulis ucapkan kepada Sekolah Vokasi IPB yang telah mendanai penelitian ini. Selain itu juga kepada keluarga dan rekan-rekan penulis yang telah membantu baik secara moril maupun materiil. Semoga penelitian ini bermanfaat untuk masyarakat.

REFERENSI

[1] J. Arifin and M. Z. Naf'an, "Verifikasi Tanda Tangan Asli Atau Palsu Berdasarkan Sifat Keacakan (Entropi)," *J. Infotel*, vol. 9, no. 1, p. 130, 2017, doi: 10.20895/infotel.v9i1.136.

[2] S. Mertokusumo, *Hukum Acara Perdata Indonesia*. Yogyakarta: Liberty, 2006.

[3] G. Shelly and M. Vermaat, *Discovering Computers: Living in a Digital World*. Boston: Course Technology, Cengage Learning, 2011.

[4] F. Song and Z. Cui, "Electronic voting scheme about elgamal blind-signatures based on XML," *Procedia Eng.*, vol. 29, pp. 2721–2725, 2012, doi: 10.1016/j.proeng.2012.01.379.

[5] Y. Ding, B. Li, and Z. Zheng, "An electronic auction scheme based on group signatures and partially blind signatures," *Procedia Eng.*, vol. 15, pp. 3051–3057, 2011, doi: 10.1016/j.proeng.2011.08.572.

[6] I. C. Chang, H. G. Hwang, M. C. Hung, M. H. Lin, and D. C. Yen, "Factors affecting the adoption of electronic signature: Executives' perspective of hospital information department," *Decis. Support Syst.*, vol. 44, no. 1, pp. 350–359, 2007, doi: 10.1016/j.dss.2007.04.006.

[7] P. Pharow and B. Blobel, "Electronic signatures for long-lasting storage purposes in electronic archives," *Int. J. Med. Inform.*, vol. 74, no. 2–4, pp. 279–287, 2005, doi: 10.1016/j.ijmedinf.2004.04.018.

[8] W. Sholihah, S. Guritman, and H. Sukoco, "Electronic Power of Attorney Protocol by using Digital Signature Algorithm," *J. Theor. Appl. Inf. Technol.*, vol. 59, no. 3, pp. 690–695, 2013.

[9] M. Wang, "Do the regulations on electronic signatures facilitate international electronic commerce? A critical review," *Comput. Law Secur. Rep.*, vol. 23, no. 1, pp. 32–41, 2007, doi: 10.1016/j.clsr.2006.09.006.

[10] H. Saripan and Z. Hamin, "The application of the digital signature law in securing internet banking: Some preliminary evidence from Malaysia," *Procedia Comput. Sci.*, vol. 3, pp. 248–253, 2011, doi: 10.1016/j.procs.2010.12.042.

[11] H. Leitold, R. Posch, and T. Rössler, "Reconstruction of electronic signatures from eDocument printouts," *Comput. Secur.*, vol. 29, no. 5, pp. 523–532, 2010, doi: 10.1016/j.cose.2009.11.002.

[12] S. Mason, "Documents signed or executed with electronic signatures in English law," *Comput. Law Secur. Rev.*, vol. 34, no. 4, pp. 933–945, 2018, doi: 10.1016/j.clsr.2018.05.023.

[13] A. Srivastava, "Electronic signatures and security issues: An empirical study," *Comput. Law Secur. Rev.*, vol. 25, no. 5, pp. 432–446, 2009, doi: 10.1016/j.clsr.2009.05.007.

[14] C. Liu, "Security analysis of Liu-Zhang-Deng digital signature scheme," *Procedia Comput. Sci.*, vol. 32, pp. 485–488, 2014, doi: 10.1016/j.procs.2014.05.451.

[15] S. K. H. Islam and G. P. Biswas, "Provably secure and pairing-free certificateless digital signature scheme using elliptic curve cryptography," *Int. J. Comput. Math.*, vol. 90, no. 11, pp. 2244–2258, 2013, doi: 10.1080/00207160.2013.776674.

[16] N. Tiwari, "On the security of pairing-free certificateless digital signature schemes using ECC," *ICT Express*, vol. 1, no. 2, pp. 94–95, 2015, doi: 10.1016/j.icte.2015.12.001.

[17] A. S. Kittur and A. R. Pais, "Batch verification of Digital Signatures: Approaches and challenges," *J. Inf. Secur. Appl.*, vol. 37, pp. 15–27, 2017, doi:

10.1016/j.jisa.2017.09.005.

[18] V. Andrianova, "Electronic signature key storage," in *Procedia Computer Science*, 2018, vol. 145, pp. 59–63, doi: 10.1016/j.procs.2018.11.010.

[19] G. Krylov, A. Gaybatova, V. Davydenko, and A. Grigoryan, "Integration of distributed ledger technology into software electronic signature exchange service," in *Procedia Computer Science*, 2020, vol. 169, no. 2019, pp. 479–488, doi: 10.1016/j.procs.2020.02.221.

[20] J. H. Seo, "Efficient digital signatures from RSA without random oracles," *Inf. Sci. (Ny).*, vol. 512, no. xxxx, pp. 471–480, 2020, doi: 10.1016/j.ins.2019.09.084.

[21] F. Shahid and A. Khan, "Smart Digital Signatures (SDS): A post-quantum digital signature scheme for distributed ledgers," *Futur. Gener. Comput. Syst.*, vol. 111, pp. 241–253, 2020, doi: 10.1016/j.future.2020.04.042.

[22] A. Alyushin, "Document protection technology in the digital economics using cognitive biometric methods," in *Procedia Computer Science*, 2020, vol. 169, no. 2019, pp. 887–891, doi: 10.1016/j.procs.2020.02.147.

[23] M. González-Lee, M. Nakano-Miyatake, H. Pérez-Meana, and G. Sánchez-Pérez, "Script format document authentication scheme based on watermarking techniques," *J. Appl. Res. Technol.*, vol. 13, no. 3, pp. 435–442, 2015, doi: 10.1016/j.jart.2015.07.010.

[24] J. C. C. Ibarz, "Bringing JSON signatures to ETSI AdES framework: Meet JAdES signatures," *Comput. Stand. Interfaces*, vol. 71, no. February, p. 103434, 2020, doi: 10.1016/j.csi.2020.103434.

[25] V. L. Xuan and D. L. Hong, "A new digital signature scheme based on the hardness of some expanded root problems," in *Procedia Computer Science*, 2020, vol. 171, no. 2019, pp. 541–550, doi: 10.1016/j.procs.2020.04.058.

[26] R. S. Pressman, *Rekayasa Perangkat Lunak: Pendekatan Praktisi*, 7th ed. Yogyakarta: Penerbit Andi, 2012.

[27] F. Sulianta, *Teknik Perancangan Arsitektur Sistem Informasi*. Yogyakarta: Penerbit Andi, 2017.

[28] R. Sukamto and M. Shalahuddin, *Rekayasa Perangkat Lunak (Terstruktur Berorientasi Objek)*. Bandung: Informatika, 2013.

[29] T. A. Kurniawan, "Pemodelan Use Case (UML): Evaluasi Terhadap beberapa Kesalahan dalam Praktik," *J. Teknol. Inf. dan Ilmu Komput.*, vol. 5, no. 1, p. 77, 2018, doi: 10.25126/jtiik.201851610.

[30] A. Menezes, P. Van Oorschot, and S. Vanstone, *Handbook of Applied Cryptography*. CRC Press, 1997.

[31] D. Rachmawati, J. T. Tarigan, and A. B. C. Ginting, "A comparative study of Message Digest 5(MD5) and SHA256 algorithm," *J. Phys. Conf. Ser.*, vol. 978, no. 1, 2018, doi: 10.1088/1742-6596/978/1/012116.