Remote e-Signing via the Web

Send and Sign Documents Online. Anywhere. Anytime. On Any Device.

NAMIRIAL GmbH

Legal Office: Seilerstätte 16, 1010 Wien, Austria
Main Office: Haider Straße 23, 4025 Ansfelden | Phone: +43-7229-88060 | www.xyzmo.com
Fiscalnumber 09 258/9720 | VAT-ID: ATU70125036

Abstract

Maximising the value of digital transformation is important in most industries—and critical for survival against competition for some. Organizations that sell virtual rather than physical products, especially, have a cost base that is largely focused on development, marketing and distribution as well as servicing which makes such organizations highly sensitive to digital transformation. For these, there needs to be a concerted focus on automating core activities to boost self-service and “straight through” transaction processing.

With online e-signing it's easy to get documents signed on their own devices without having to meet the recipient in-person. Simply send documents links out for signature to other people, get instant visibility into your document status, access completed documents, and much more. Whether you or your recipients are in the office, at home, or on the go, online e-signing works every time from every device.

If you send documents out for signature, the recipient gets an email with a link to your document and can sign on a smartphone, tablet, or any web/HTML5-enabled device without the need to download anything. You can have multiple signers and get them to sign in the order you need. E-signatures don't just let you reach customers on the devices they most commonly use; they also let you create compliance and comfortable engagement. E-signature solutions allow you to build in markers (tags) and metadata about documents that can help consumers understand what they are signing and what fields they have to fill out. To eliminate human error, electronic documents can also include auto-check tools that identify common mistakes in time. All these functions together add up to a better quality of the contracts and customer experience.

This white paper will help you to understand what a remote online e-signing solution needs to provide. With an emphasis on why the last mile to the signer needs to be closely managed and what this includes, it will help you to select the most appropriate methodologies for authenticating a remote signer. Then we will discuss the most fitting use for each signing technology—biometric, HTML5, or certificate-based signing—and why this is, depending on the actual use case. There you will see that e-signing is about more than simply signing digital documents—it’s about optimizing the whole process. Finally, we also illustrate the end-to-end business process that a real-customer (The Phone House) has implemented.

Table of Contents

1 Typical Functionalities

To efficiently process document based transactions with remote recipients via internet, online signing solutions typically consist of the following three basic functionalities:

- Definition of the transaction (Set up a signing-envelope)
- Complete the transaction (complete the contract and signing)
- Reporting.

Each task has certain typical steps, as outlined below. Additionally the system has to offer administration features to manage the users and templates, and adaptable branding if required.

1.1 Definition of the transaction (signing-envelope)

Actions performed by the sender:

- Start a new “envelope” (a container used to send one or more documents for signature) or use a template to start with.
- Add the required documents.
- Add recipients. This could be signers themselves or recipients of a copy.
- If you are uploading a PDF document with form fields, the fields are automatically detected.
- Place tags/markers in the document for signatures, attachments and other information.
- Add your subject and email message.
- Set recipient options, reminders, expirations, and more.
- Send the envelope for signature to all configured recipients.

These important steps are explained in detail in section 2 - Managing the Last Mile.

1.1 Complete the transaction (signing)

Actions performed by one or more recipients:

- Signers receive an email with a link to the document.
- They click the link.
- There is typically no need to download or sign up for anything.
- They may have to further authenticate themselves, when required by the sender.
- They can review and print the document or complete form fields and add attachments.
- Whenever they are ready, they execute a signature field using a stylus by hand (“Draw to sign” – on their smartphone or tablet), or they "Click to sign" or "Type to sign" from any web-enabled device including PCs with just mouse and keyboard.

The possibilities and requirements of the most important actions in this step of the signing process is described in detail in the following sections:

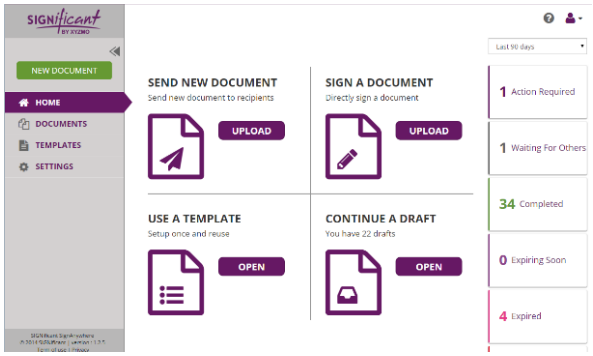
- User (signer) authentication (see section 3)
- Signature execution (see section 4).

1.2 Reporting

Efficient reporting should help the sender to pay attention on those transactions, which still require actions to successfully finish them. Such transactions are documented and recorded in order to prove them anytime it is necessary. Therefore a simple overview is essential:

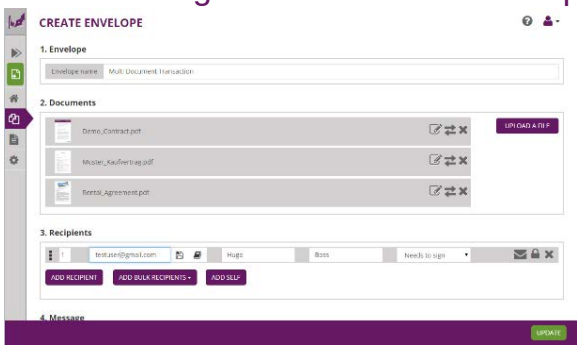
- Dashboards help to get an overview over all transactions.
- With a detail view you can easily check the status of every envelope. Thus you always know where the document is stated in the signing process.
- Easy access to the audit trails which is required for legal proof if needed.
- Set reminders to notify yourself or your recipients at each important step of the process.

2 Managing the Last Mile



Especially in online scenarios where the document recipients needs to sign remotely without a physical face-to-face meeting, it's vital that the e-signature software ensures proper process execution, maximizes automation of all steps, and only raises alerts and reminders if something goes wrong. This can dramatically reduce the work-effort involved in getting documents signed as, typically, more than 80 percent of all signature transactions do not require any manual interaction of the sender.

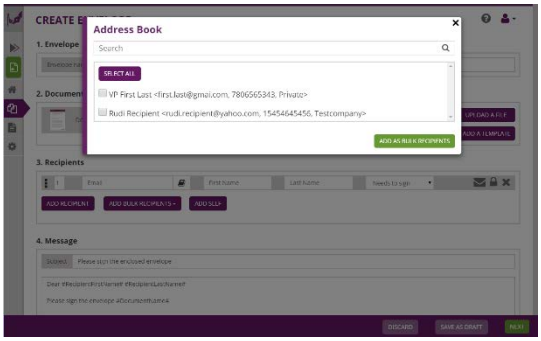
1.3 Creating transactions with multiple documents



Often, a transaction consists of signing multiple documents. Using an "envelope" concept, you can define which documents belong to a single transaction and thus need to be signed together to successfully accomplish this.

Allowing you to create and reuse envelope templates makes it easy to build on work that already has been done previously, potentially saving you a lot of time.

1.4 Routing / Workflow



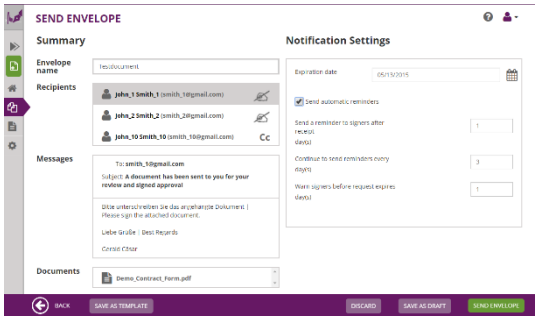
With routing rules you can define the order of persons that need to sign a specific document and should have access to it once it is completed—either as a signed original or as a flattened copy. This may be done either in parallel, or sequentially whereby you define the precise order in which each recipient will receive the envelope.

Additionally, you also can send the same documents to a large number of (a bulk of) recipients. Once you send the envelope, a separate envelope for each recipient is created. This is especially useful if e.g. a large group of people need to sign the same document. A typical example is a policy document that all employees must sign.

1.5 Defining document ceremony per recipient

Here, you simply define online in a web browser what each recipient has to do to successfully complete his/her part of the transaction—be it filling out a form field, adding a particular attachment, or signing signature fields. Alternatively, you simply define these tasks automatically from external applications through API calls, or use text markers to define signature fields within a document.

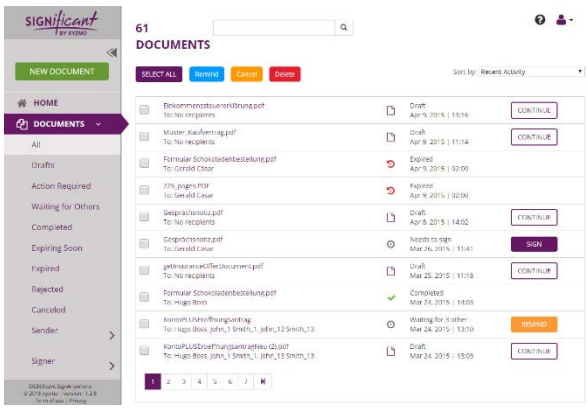
1.6 Reminders and alerts

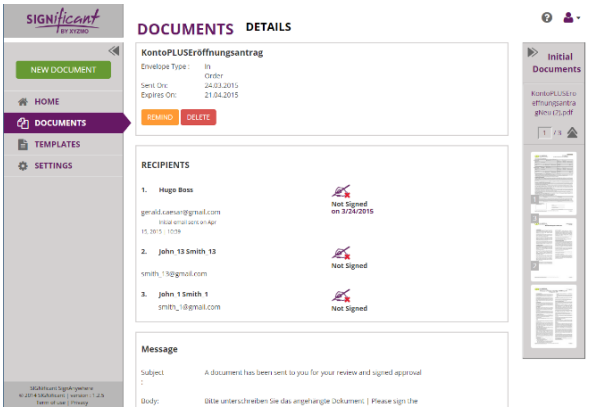


With reminders, you can define rules to remind your recipients of their signing tasks and to expire documents if they are not signed within a certain time frame. Using alerts, you can remind the sender that a specific document is not yet signed or if the recipient refuses to sign a document

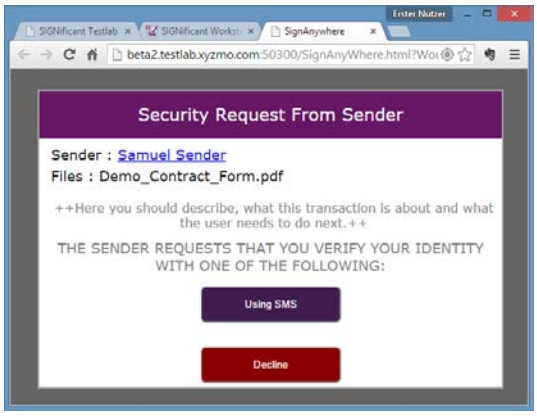
1.7 Dashboards

Dashboard views are important to give a quick overview about the status of your envelopes. Good dashboards not only provide an overview, but also allow a fast search through and editing of the areas that require your attention.





3 Authentication Methods



To protect a document (envelope) from unauthorized remote access via the web, it is critical that the online e-signature solution first authenticates accessing users. Depending on the relevance of the transaction there are different possibilities e.g.:

- The fact that the e-mail including a link to the document, sent to the receiver is enough.
- The receiver has to enter an access code, which he got via another channel.
  - Apply existing methods that you are already using or the receiver is getting the link just by entering a secure area with his login data.
  - Additionally the receiver has to authenticate himself via Facebook and other social media platforms.
  - The receiver gets a one-time PIN sent to a pre enrolled number of his mobile phone and has to enter this code within a certain time.
  - Authentication with national ID card or passport.
  - Authentication and digital signature with third-party signing certificates.

Logging the executed authentication into the secure audit trail of the then signed document eventually enables you to reliably prove that only the previously identified user was able to actually sign the document in question.

1.8 Email authentication

This scenario is typically suited to HTML5 signatures for B2B scenarios and non-business-critical documents for which you want to make signing as easy as possible. You simply send the link to the inbox of the recipient; no further authentication is necessary. In case of a dispute, you can prove with the audit trail that you sent the document to a specific email address, and often you have the IP-address of the computer and geolocation as well, if the recipient did not explicitly block it.

All of the following authentication methods start with this scenario and add additional authentication steps on top. Although arguably not quite as robust as a full biometric signature, each additional step below adds a little bit more evidential weight in case of dispute, making them good enough for many use cases. Strict methods authenticating the receiver without any doubt also allow implementation of the advanced electronic signature. You even may go further and require the use of qualified electronic signatures (QES), however resulting in a limited number of potential users. An overview is given in the following sections.

1.9 Recipients required to enter an access code

In addition to the above, you can present the recipient with a security request page and require entry of a code to access the documents for viewing and signing. After the access code has been entered correctly, the recipient is led through the normal signing process. The access code is not sent in the same email that includes the link, as this would make it totally unsecure. Typically, the code is not even sent by email at all, but instead is communicated via another channel such as by phone. To have a code that works for a longer time, you can choose to have the access code agreed upon between the parties in a separate process.

The value of this is in case of a dispute: The sender can prove that the signer must have had access to the access code in order to sign the document.

1.10 Leveraging trusted authentication models that you already have in place

Some businesses have customer portals or other software already in place and the user is identified by such systems. Let’s assume the user is within a banking application where he/she manages all his/her transactions. If the user has to sign a document and has already been properly authenticated, then there may be no need for further authentication. But the proof of authentication has to be included in the audit trail in any case.

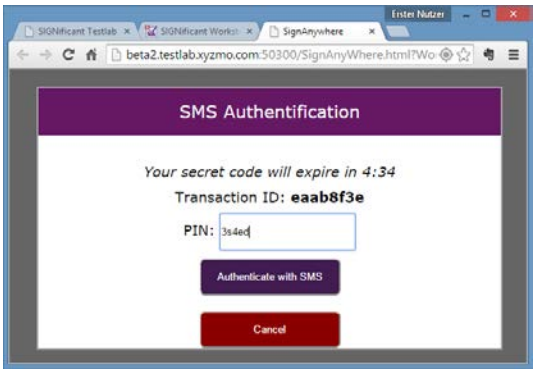
Another scenario is one in which the recipient already uses a secure authentication method (e.g., a token) for other purposes and this infrastructure is repurposed to authenticate him/her for document signing. As above, the authentication process must be included in the audit trail of the signed document to provide evidence.

1.11 Using social networking IDs such as Facebook login

With this scenario, popular social networking sites are used for authentication. Most of them provide services to allow third-party applications to authenticate users. The quality of this authentication depends very much on the quality of the social network profile that is used for authentication. In addition, the quality and amount of the data that you receive about the

user from the social network for the audit trail has to be considered as this can be restricted to very basic information types.

1.12 Sending an SMS with a one-time password



This authentication method capitalizes on the fact that mobile devices provide, worldwide, a very good means of identification for their owners. Many people carry mobile phones throughout the day and have them within reach at all times. The mobile telephone number is a strong identifier for the owner. This is why many banking and other applications already use this method for online transactions.

Additional security can be added if the one-time password is time-limited, e.g., if the recipient has to enter it within the next 5 minutes.

Another consideration is whether the one-time password should be valid only for one signature or for the entire process.

In addition, it's recommended to include a unique identifier for the transaction in the SMS sent to the recipient together with the one-time password. This unique identifier should also be displayed in the security dialogue where the recipient has to enter the password, to allow him/her to prove that the password is for this transaction alone and not for another.

All of these messages must form part of a proper audit trail for the purpose of being able to prove everything that has been exchanged in the case of a dispute.

It's worth considering how the sender should be able to handle and define the one-time password settings. There are three main options to select from here:

1. The recipient can enter his/her mobile phone number himself. Clearly this is convenient for both the sender and the recipient, but it invites potential misuse by the recipient.
2. The sender defines in advance which mobile number has to be used and the recipient cannot change that. This scenario is very common as it adds considerable security to the process.
3. Finally there are scenarios where both parties—recipient and sender (e.g., a sales employee)—should have no chance to define (change) the mobile number for the recipients and often do not even see them. In this scenario, the sender can only select the recipient from a list of names. The mobile numbers are stored in a central place and cannot be adjusted by the sender. This can even be taken a step further with an upstream process in which the recipient agrees, for example when visiting the branch, in writing that in the future, a one-time password sent to his predefined mobile number can be used as an equivalent to his signature.

Certain countries assign the same legal value to a proper implementation of this method—or special, country-specific versions of it in conjunction with certificate-based signatures (see Section4.2)—as a handwritten wet-ink signature, an example being "Handysignatur" in Austria.

1.13 Authentication with national identity cards or passports

Many countries supply their citizens with electronic identity (eID) cards that provide the machine-readable travel document functionality specified by the International Civil Aviation Organization (ICAO). In Europe, most ID cards are based on the European Citizen Card (ECC) specification and not only print the cardholder's personal data (e.g., name, date and place of birth, nationality) on the face of the card, but also store the data in the integrated chip. Cardholders can use this eID function to carry out legally recognized transactions with public authorities and private companies via the internet.

Germany's new identity card (nPA), for instance, achieves access control of the personal data stored on the card through a Password-Authenticated Connection Establishment (PACE) protocol and Extended Access Control (EAC). The PACE protocol performs user authentication using a Personal Identification Number (PIN) and establishes a secure connection between the ID card and the card reader to protect the communication across the contactless interface. The cardholder gives his/her consent for card access by entering the PIN.

1.14 Authentication and signing with third-party digital certificates

In cases where the recipient already has in place a Public Key Infrastructure with personal digital signing certificates ("PKI", for example on smart cards, USB-tokens, purely as software on computers), this certificate can be used not only to authenticate the signer, but also to digitally sign the document. Most often, it is simply an attempt to reuse the existing PKI infrastructure, which was in place for other purposes. This allows you to take advantage of the existing certificate and use it for the digital signing of the document.

Some national identity cards even go beyond the pure eID function and also directly provide the function of using it to execute a qualified electronic signature (QES). Although, theoretically, this is a great way to reuse infrastructure that is already in place, market penetration and user acceptance is often problematic, as described in section 4.2.

4 E-Signing Technologies

There are three different technologies to digitally sign a document. First, there is an important difference between methods in which:

- the captured handwritten signature of a person is forensically identifiable (also known as a "biometric signature"),
- the embedded HTML5 signature in the signature field (e.g., image of handwritten signature graph) is not sufficient to authenticate the signer, making additional authentication methods and audit trails necessary to be legally binding,
- signatures are used in conjunction with personal digital signing certificates.

Thus, the main question in capturing handwritten signatures is whether the captured signature data is forensically identifiable. One can say that in all scenarios featuring the use of a stylus or even a pen provided by the vendor of the device and proper

implementation of the capturing software, the result will be signatures that are forensically identifiable.

In other scenarios such as signing with a mouse, touchpad or finger—or where the necessary capturing software and/or hardware is not in place—the signature is not forensically identifiable. This second category is what we'll call an "HTML5 signature."

Certificate-based signatures, by contrast, require a PKI infrastructure, and while they are a very popular model for e-signing within your own organization (if there is already a PKI infrastructure available), they can provide only limited penetration in any other scenarios, such as a B2C or B2B contract.

Regardless which of those three signature methods is used, the signed document and every single signature in it should always be sealed with a valid digital signature to ensure their validity.

1.15 HTML5 signatures

The big advantage of HTML5 signatures is that they do not require the signer to install anything. They are simply formatted to work on any HTML5-enabled web device. Depending on the authentication method (see section 3), they also do not require complex sign-up procedures, so they are perfectly suited to online B2C and B2B scenarios.

However, the whole process is fully dependent on the proper authentication of the recipient (see section 3) and the logging of all user interactions. If this is securely documented in an audit trail, then the HTML5 signature provides reliable evidential weight. Depending on the chosen method it may even fulfill[1] the EU's advanced electronic signature standard and thus be fully equivalent to a forensically identifiable (biometric) signature, which is described in section 4.3.

Furthermore, a proper audit trail that is sufficiently easy to be read and understood by a judge and involved lawyers—and that doesn't force a judge to go for an expert opinion—places the burden of proof immediately on the signer in most cases, which even offers an advantage over biometric signatures, which are not verified in real time.

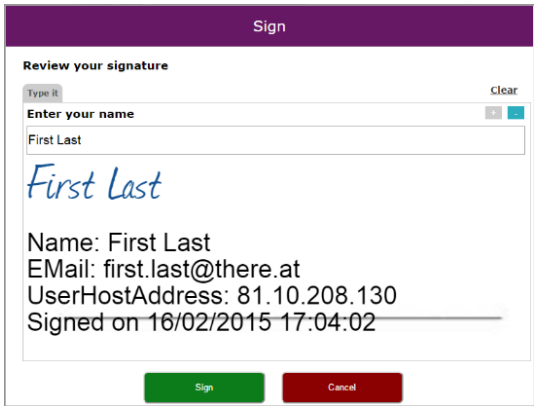
The question of how this signature is displayed on the document is more a question of convenience for the signer and isn't primarily a legal question. Maybe one can argue that if the signer selected or constructed the signature image himself—by, for example, typing the name—it has more legal weight compared with methods where that's not the case.

[1] Voithofer, Paul – Gutachterliche Stellungnahme SignAnyhwere [2015]

1.15.1 Click-2-Sign

This is somewhat the equivalent of the stamp imprint in the old paper world. Proper e-signature software will allow you to define the elements of the stamp imprint. Depending on the use case, you might only want to include the name of the signer, or also the IP-address, geolocation, and other information. You may even want to add a text that states this is an electronic signature and not a real one.

1.15.2 Type-2-Sign (Typing the name)



This method gives the option of entering the name and using various handwritten fonts to convert the name into a placeholder that looks like a handwritten signature. Users may choose the font and the screen size they prefer.

Similar to the Click-2-Sign, the Type-2-Sign signature also may include additional informatioan in the imprint, like the signer's name, email, IP address, and signing date & time.

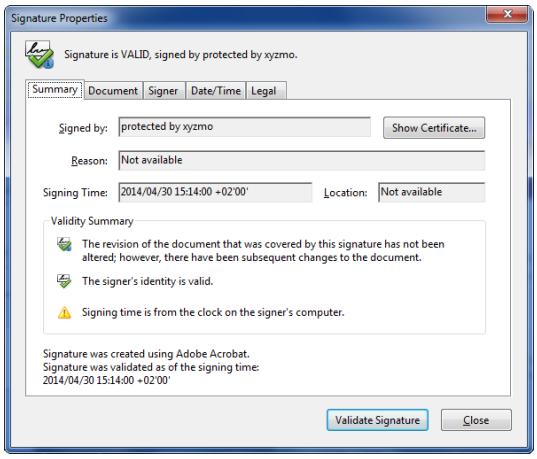
1.15.3 Draw-2-Sign (Drawing the name with a finger, mouse, or stylus)

The signer draws his signature as he is used to doing on paper. This is similar to methods where you try to capture the real signature, but typically people are not able to draw their signature with a finger and most people definitely cannot do so with a mouse. Also, even if a stylus is used, the signature image is not forensically identifiable as HTML5 based solutions cannot capture any reliable biometric data, only an image. Therefore, the separate authentication step is still necessary.

1.16 Certificate-based personal signatures

Some industries and a number of countries demand certificate-based personal digital signatures. In this case, senders need to be able to require signers to apply digital signatures with signing certificates that are issued to them "personally."

The process is very similar to the standard process, thus:



- A new envelope is created and documents are added to the envelope as normal.
- Recipients are added as normal, but the sender requires to apply a digital certificate for some recipients.
- Any other authentication options for the recipient are added as normal.
- The configuration of the envelope is completed and it is sent as normal.
- The recipient opens the envelope and adds information in all the required fields as with all other methods.
- When the signer executes a signature the document is signed with the signers personal digital signature certificate that he or she must possess (e.g. on a smartcard or token) and where just the signer holds the password for.
- The signer is asked to review and confirm the information, maybe including the reason for the signature and his/her company details and location.

After these steps, everyone can inspect the digital signature in a popular PDF reader.

As with all technology, criminals and fraudsters will try to find ways of circumventing the intended process. For example, in the case of a personal digital certificate through manipulation of the card reader.

However this method isn't completely doubtless, in some countries the highest legal value of a signature—which is deemed to be equivalent to a "wet ink on paper" signature—can only be realized by using such certificate-based signatures. Often they even require the use of approved chip cards and reading devices, which renders this technology quite expensive and complex. This generally applies to the European Union with its so-called Qualified Electronic Signatures (QES). Some member countries even provide an infrastructure to activate the QES function on their national identity cards (e.g., Germany with its nPA; see also Sections 3.6 and 3.7). However, typically, card owners have to activate and pay for this function separately and also require a card reader to use it, which results in low market penetration, which makes its use problematic in B2C scenarios.

1.17 Forensically identifiable signatures (biometric signatures)

A forensically identifiable signature is much more than merely a digitized image of a handwritten signature. It requires recording the handwritten signature of a person using all available parameters, such as acceleration and speed—i.e. the writing rhythm. These dynamic parameters are unique to every individual and cannot be reproduced by a forger. That's why the digitized signature is forensically

identifiable (and far more reliable than with the signed image alone).

When someone claims “I didn’t sign that,” a forensic expert can always perform a thorough manual signature verification at any time afterwards, using specialized software to achieve an admissible result in the same way as the expert would with a signature on paper. Thus, the biometric signature fulfills[2] the EU’s advanced electronic signature standard and has been widely adopted as the de-facto industry standard wherever it is applicable.

Some solutions also provide a signature verification that authenticates a signature against a pre-enrolled signature profile database in real time. This allows you to not only secure the execution of certain transactions, but also to provide a ready-to-use audit trail in case of a dispute, thus placing the burden of proof immediately on the signer.

So, why not use biometric signatures all the time?

The problem is that their reliable capturing requires a real-time environment on the computer, tablet, or smartphone that is used to record their dynamic aspects. This can only be provided by a native local software component or a Java applet/browser-plug-in in a web application, but not with pure HTML5 alone. Additionally, secure encryption is key when capturing biometric signatures, which is again something that can only be provided through a local software component, but not through HTML5, as the capture logic’s source code is always visible to the client and thus can easily be replaced by an unsecure source code through injected JavaScript code.

However, requiring the signer to install a local signature-capturing component is, in many situations, not a practical approach, which is why HTML5 signatures also have a very wide use case. However, wherever possible, or for high-value or high-risk transactions and commitments, it is best to rely on handwritten biometric signatures.

1.17.1 Capturing devices for biometric signatures

On the one hand, there are the traditional signature pads and pen-enabled screens, while on the other, there is a broad selection of smartphones and tablets that have native pen support. In addition, there are special pens that allow very good signature capturing on devices that have no pen support out-of-the-box, such as the iPad or iPhone. Many of these special pens even deliver pressure values, and some promise palm protection, but in many cases the palm protection and data rate are not as good as with native pens. However, if you do not have a native pen, you still can use a capacitive stylus, as discussed below.

[2] Voithofer, Paul – Sachverständigengutachten SIGNificant Produkte [2012] & Caspart, Wolfgang – Graphologisches Gutachten [2012]

a) Stylus

Signing with a capacitive stylus gives you the feeling of signing with a pen. There are still a few shortcomings compared with signing with a native pen, which typically results in larger signatures that are written at a slower speed. However—in contrast to signing with a finger— the captured signature and writing rhythm is consistent and similar enough to the process on paper, which makes it valid in case of dispute. Additionally whenever it is possible the signature is compared to pre enrolled signatures, which were taken with similar technologies – preferred with the same stylus.

b) Native pen

Native pens typically provide a signing experience that is, compared with a capacitive stylus, even closer to the act of signing in the paper world.

The reason for this is that native pens provide:

- A thin pen tip, like your ink-to-paper pen, that enables you to sign with your regular small letters
- Palm protection so that you can touch the screen while signing without reducing the quality of the captured signature.

Additionally, native pens also provide a better data quality because:

- They provide a higher data rate, allowing you to capture all aspects of even very fast signatures
- Many also capture the pressure information of your writing, which—while not mandatory for capturing a biometric signature—adds extra security and evidence as it provides additional signature data that a forensic expert can analyze.

c) Fineline stylus

A fineline stylus aims to bring the advantages of a native pen to devices that do not provide an out-of-the-box stylus, the most prominent example being the iPad. It is still a capacitive stylus, but one that uses electronic technology to allow usage of fine pen tip, and sometimes also for palm protection and pressure recording. While it will not be as good as with a native pen, the writing experience is certainly better than with an ordinary capacitive pen. Also, pen technology is constantly improving, so we will see increasingly better fineline styluses in the future.

1.17.2 Using a smartphone as a signature pad

This scenario is perfect for those instances in business when you want to capture biometric signatures, but do not want to deploy signature pads or pen displays.

The typical process is as follows:

- Review documents or complete form fields and add attachments on any computer in a browser—maybe together with a customer, employee, or business partner—and use a smartphone as a signature-capturing device.
- A native signature capture app turns a smartphone into a signature-capturing device. This app should be available on most iOS, Android, and Windows phones.
- When the signer is ready to sign a document, a secure communication between the smartphone and the host computer is established using a token (which you may read simply by using the smartphone's built-in camera using a QR-code reader integrated into the native signature capture app).
- The signature capture app shows a signature capture dialogue, with the document background providing a visual document mapping.

- The signature is captured on the smartphone. It's highly recommended to use smartphones with native pens or a stylus for signing, otherwise you may lose the potential for forensic identification.
- After the signature is captured, it's transferred via the secured channel and embedded into the document.

5 Platform Aspects

Many important requirements to online e-signing can be shared with other use cases (e.g., POS or mobile), which is why we have discussed them in our platform white paper "Get Documents Signed. Anywhere. Anytime."

This includes the following aspects:

- Security
- Long-term archiving
- Process evidence (incl. audit trails)
- Deployment methods
- Enterprise integration
- Standard versus proprietary approaches.

As we do not want to reiterate these general aspects, we recommend that you read about them in the above-referenced platform white paper.

6 SIGNificant-References

SIGNificant provides an enterprise e-signature platform that allows you to conveniently send documents for signature or simply sign them yourself online. SIGNificant efficiently provides you with the user interface and tools needed to define an optimal e-signature process and user experience.

Whether for HTML5 signatures, biometric signatures, or digital signatures with personal certificates, the platform's building blocks make it easy to choose the best combination of signing method and signer authentication, regardless of which signature device the recipient uses.

To better illustrate how SIGNificant can be applied in selected industries for online signing scenarios, the following section outlines a real case study of a customer that implemented SIGNificant for online-signing with its end-to-end business process implemented.

6.1 The Phone House Netherlands

Use case:

- Digitally sign insurance contracts for mobile phones online in the web browser on any HTML5 device

Deployed products:

- Signing application: SIGNificant Server with SignAnywhere

End-to-end business process:

1. The client selects the insurance contract they want to add to their phone online on The Phone House website.
2. The Phone House backend systems automatically create the insurance contract to be signed and send a link to the contract to be viewed and signed online to the client.
3. The client opens the insurance contract in a browser that supports HTML5 and signs two signature fields through Type-2-Sign or Draw-2-Sign.
4. The signed document and the audit trail are safely stored in The Phone House's archiving system.
5. The client is informed about the results of the transaction online and can access a copy of the signed insurance contract directly in their web browser session.