Hellosign.com Feature List

Version 1.0

www.mindts.com

SR. No.	Table Of Contents
1	ABOUT CLIENT
2	WHY US?
3	MINDTS PROPOSAL
4	OUT OF SCOPE
5	MINDTS WORK APPROACH
6	PROJECT DELIVERABLES
7	CONTACT DETAILS

1. ABOUTCLIENT

Client	About Client
Dani	

WHY US?

Full Life-Cycle Development Services:

MindTS is a full life cycle software development and services company providing premium services to its clientele across the globe. MindTS is quite a young company (founded in April 2013) and has been providing software development, web development, web design, custom application development, product development, rich internet applications and design / multimedia services. MindTS provides a quality driven approach towards software development and provides end-to-end customized solutions that meet client requirements.

Vast Pool of IT Professionals:

We have a strong team of over 20+ experts consisting of business analysts, programmers, team leads, project managers, testing and quality professionals, GUI specialists and product specialists to cater to your requirements.

Quality Approach:

MindTS adheres to strict quality measures. Importance of quality applications is paramount at MindTS and each project is constantly monitored and controlled by quality assurance professionals.

Client Centric Approach & Focus:

We are the best in our industry because we listen. We make attempt to understand our clients' objectives and needs, and we collaborate to find solutions that meet them or exceed them.

Timely Delivery:

Timely deliverance: Thanks to our experience, our talented programmers, testing tools, and effective project management, we can implement and deliver your projects on time. "Accountable, Creative, Simplistic &Systematic."

MINDTS PROPOSAL

Whether building a web site, or an application, it is critical to clearly define the requirements for that site or application. Otherwise the site or application will not operate as needed to meet the business objectives. There are four categories of requirements that need to be documented: business, content, functional and technical.

Scope:

The scope of this project is deemed to include the requirements gathering, design, development and testing of all elements of the project. The scope of the project covers the following:

Requirements gathering and refining

Functionality specification and/or prototyping

Technical specification

Development of core site and functionality

Testing of site functionality

Provide 1 month's FREE support for bug fixing after the site goes live

Project Overview:

Here, we would like to explain the standard operation process for this project.

MindTS considers the project completion in Five Phases:

- **Project Planning/Requirements Analysis**
- **Design**
- **Development**
- **Quality Testing**
- **Final Deliverance with Technical Support**

Project Planning:

Requirement Analysis refers to information gathering, we collect the information, goal websites you like or else any reference website which you were suggested for the project.

Design Analysis:

Here in this phase we will send you some sample design for the respective website. You can select the design pattern with color pallets, background design, and responsive design (It comes under HTML/CSS Part, which works in mobile, IPad etc. Database Design defines the different informative tables; we create such tables for reference categories.

SCOPE ANALYSIS:

The purpose of the project to develop and Web application similar to Hello Sign. We will be developing all the features of Hello sign along with API development. Our Application is an electronic signature tool that enables users to sign, fill out, send, retrieve, and save documents paperlessly. Users may fill out and sign documents on a computer, tablet, or mobile phone. Users may also upload and prepare documents to be signed and filled out by another party or parties, otherwise known as a "signature request." Recipients receive the signature request via e-mail, in which they click the "Review & Sign" button to begin the signature process. The document will display, and signers are guided through the document indicating where they need to sign, fill in text, or select a check-box, as indicated by the originator of the document. At the end, the signer clicks "I Agree," and a copy of the signed document is e-mailed to the signer and returned to the requester. Users may also create and save editable templates for documents they send repeatedly. Users may add co-workers to their "team" and share templates with team members as well.

BASIC FEATURES

1.REGISTRATION/ SIGN UP / LOGIN.

2.ELECTRONIC SIGNATURE CREATION.

3.PAYMENT.

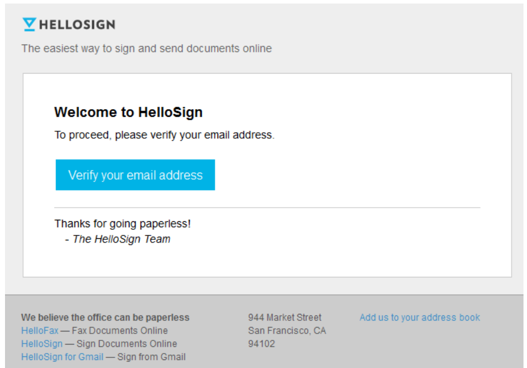
4. MY ACCOUNT.

5. API

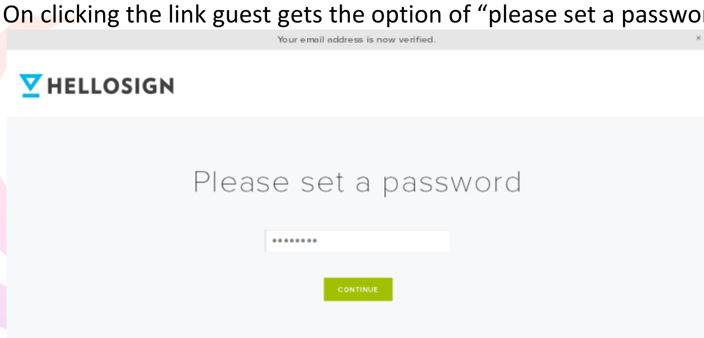
User registration –

1. There are 2 ways to login in similar to hello sign –
   1. **Through Email.**
   2. **Google plus.**

1. **<u>Through Email: -</u>** Guest needs to type his verified email address and after clicking "sign up free" he is taken for tour within the website where guest can explore the application and services provided by them. And simultaneously guest gets a link for verification in his inbox of registered email address.



- ✓ On clicking the link guest gets the option of "please set a password".



- ✓ After setting password he proceeds towards home page of his registered hellosign account.

2. **Google plus**: -
   - ✓ On clicking "sign in with Google" guest is redirected to another tab where guest needs to fill in his Gmail address and then enter the password.
   - ✓ After this process, guest is able to access his registered hellosign account.

<u>Login</u>
<u>Steps:</u>

- ➢ Guest fills in email and password.
- ➢ If the entered email and password match a user in the system, the Guest becomes logged in as the new user.
- ➢ If there are no matching users, warning is shown.

Forgot Password

- ➢ Users can request for password if he has forgotten the password. They need to provide e-mail address used during the initial registration.
- ➢ If a Login Name/Password is found in the database for that particular e-mail address, then e-mail will be sent to the user with appropriate login information.

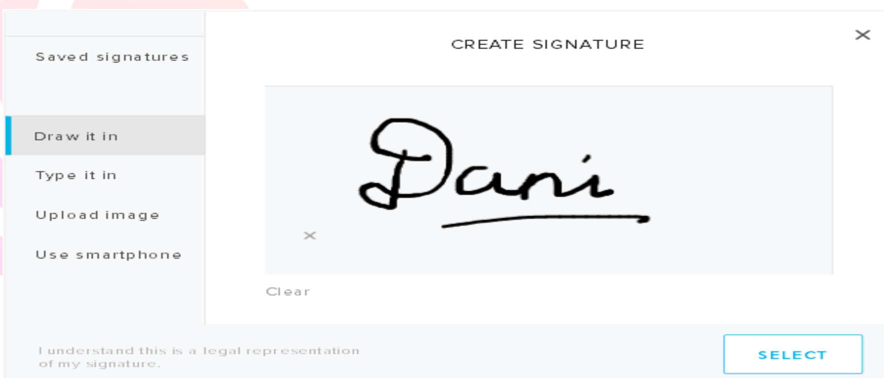
Change Password

- ➢ User can change the password after successful login.
  - Old password
  - New password
  - Confirm password

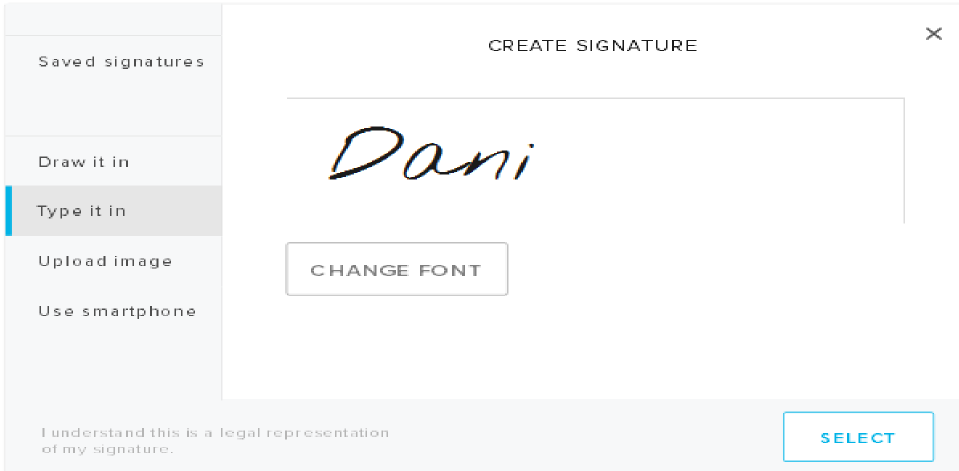
Create ELECTRONIC SIGNATURE

There are 4 ways to create electronic signature using HelloSign.

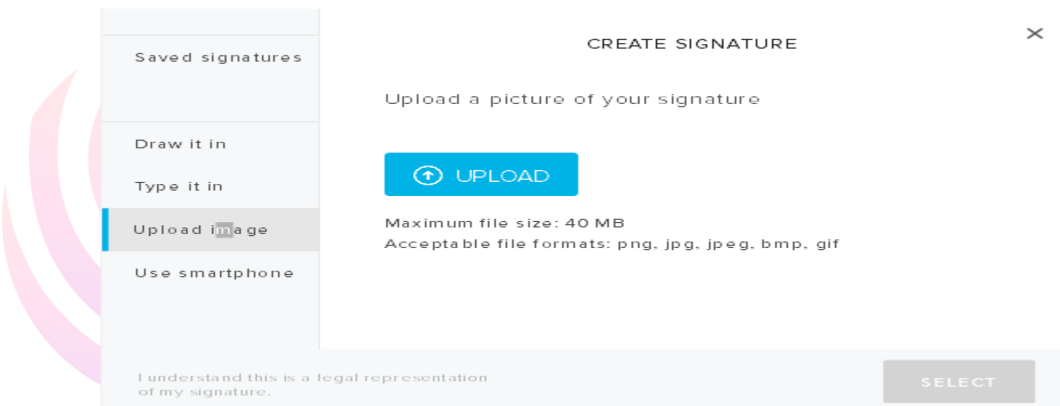
- Draw your signature using your computer mouse or touchpad.



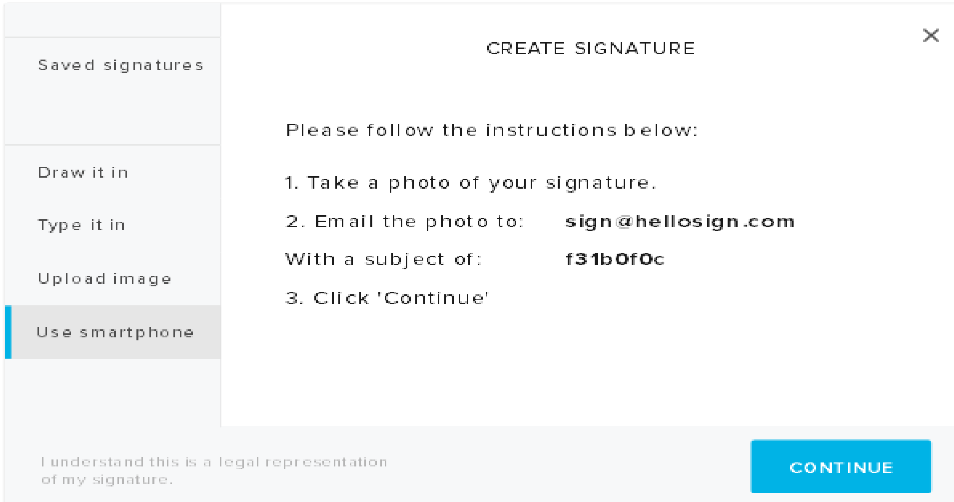
- Type your name and select any one of our fonts to give your typed signature an authentic touch.



- Take a photo of your signature using your Smartphone and upload it to HelloSign. Maximum file size should be 40 MB and acceptable files format are png, jpg, jpeg, bmp, gif.



- Sign using your finger either on our mobile app for iOS and Android, or using our in-person signing feature on the iPad.



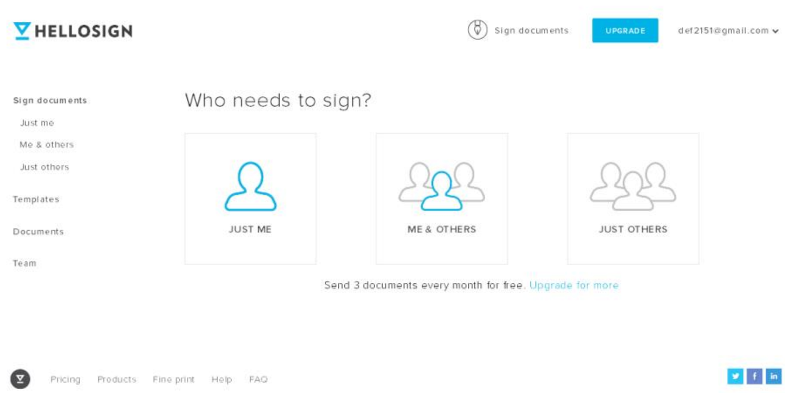
Once created, an electronic signature can be added to any document along with other annotations such as check marks or freeform text.

Login Section

- ➢ User will be asked username and password for login.
- ➢ Member will get welcome message (Ex: Welcome "Username") after successfully logging in.

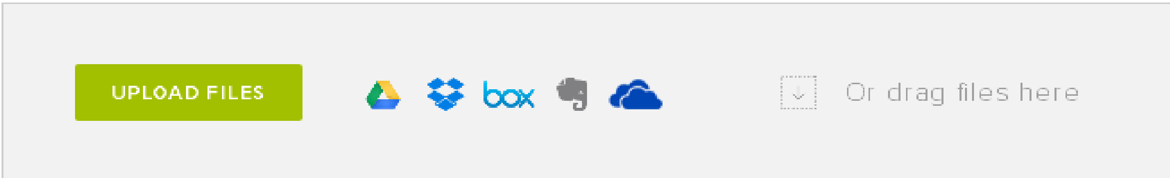
After signing in on the dashboard we get 3 options under "Who needs to sign".

1. JUST ME
2. ME & OTHERS
3. JUST OTHERS



✓ "JUST ME"

On clicking "JUST ME" user gets the option of upload files.

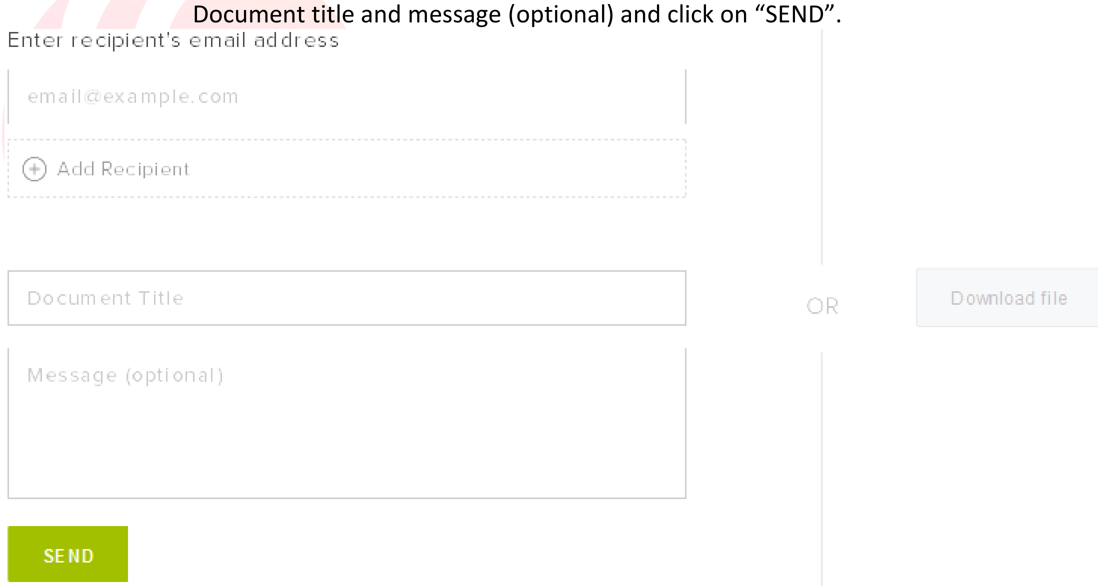


- ✓ In this option the user is able to upload files using various options like Google drive, Dropbox, Box, Evernote, OneDrive, Or user can use the option of drag and drop file from computer. To use your Google Drive files, user must install the add on and launch the application
- ✓ On clicking upload file option user can select the file of any format from the computer, and upload it.

- After uploading, the app converts the document in user friendly format, where he can put signature, date, texts and other details.

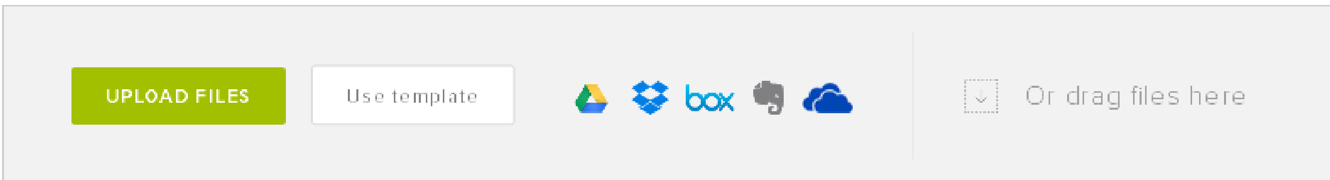
- Now click on "PREPARE DOCS FOR SIGNING". When you do this, a popup window will appear that contains your document. Look towards the top of this document at the row of buttons labeled signature, initials, textbox, checkbox, and sign date. Drag-and-drop each of these boxes to their respective place on your document. For instance, drag a signature field to the signature area on your document. If you have multiple signers, make sure to select who signs and fills out what information from the drop-down menu above the respective box.

- ✓ Then the user needs to fill in his email address, add recipients (if any), enter the Document title and message (optional) and click on "SEND".



- **ME AND OTHERS**

✓ On clicking "ME AND OTHERS" we get the option of "UPLOAD" and additional option of "USE TEMPLATE". In this option the user is able to upload files using various options like Google drive, Dropbox, Box, Evernote, OneDrive, Or user can use the option of drag and drop file from computer. To use your Google Drive files, user must install and launch the HelloSign application.

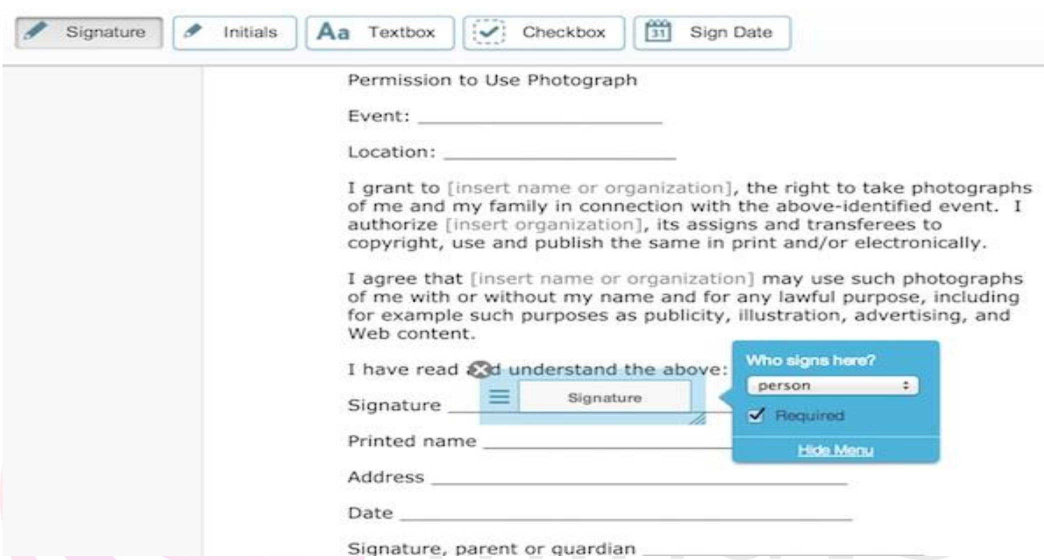


✓ On clicking upload file option user can select the file of any format from the computer, and upload it.

✓ After uploading, the app converts the document in user friendly format, where he can put signature, date, texts and other details.

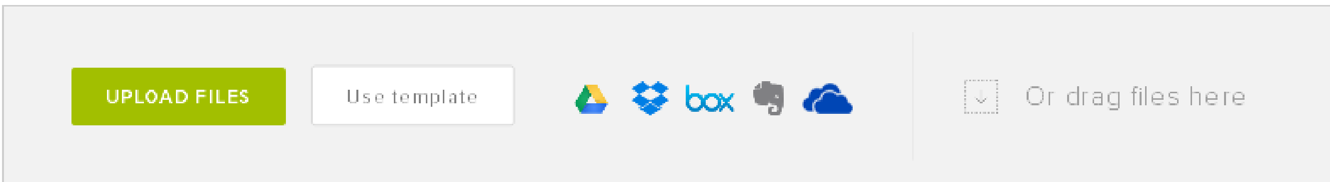
✓ In the next step user will find two text boxes where user can add his additional signer's name and email. If you need to add more signers, look below the boxes and click the + Add Signer button. User can also copy other email addresses to the form from the + Add CC button.

- ✓ Now click on"PREPARE DOCS FOR SIGNING". When you do this, a popup window will appear that contains your document. Look towards the top of this document at the row of buttons labeled signature, initials, textbox, checkbox, and sign date. Drag-and-drop each of these boxes to their respective place on your document. For instance, drag a signature field to the signature area on your document.If you have multiple signers, make sure to select who signs and fills out what information from the drop-down menu above the respective box.



- ✓ On the bottom of the page under the Add document title and custom message header, the user can write a message for signees and name the document before sending it to signees.

- **ME AND OTHERS**
  - ✓ On clicking “JUST OTHERS” we get the option of “UPLOAD” and additional option of “USE TEMPLATE”. In this option the user is able to upload files using various options like Google drive, Dropbox, Box, Evernote, OneDrive, Or user can use the option of drag and drop file from computer. To use your Google Drive files, user must install and launch the HelloSign application.

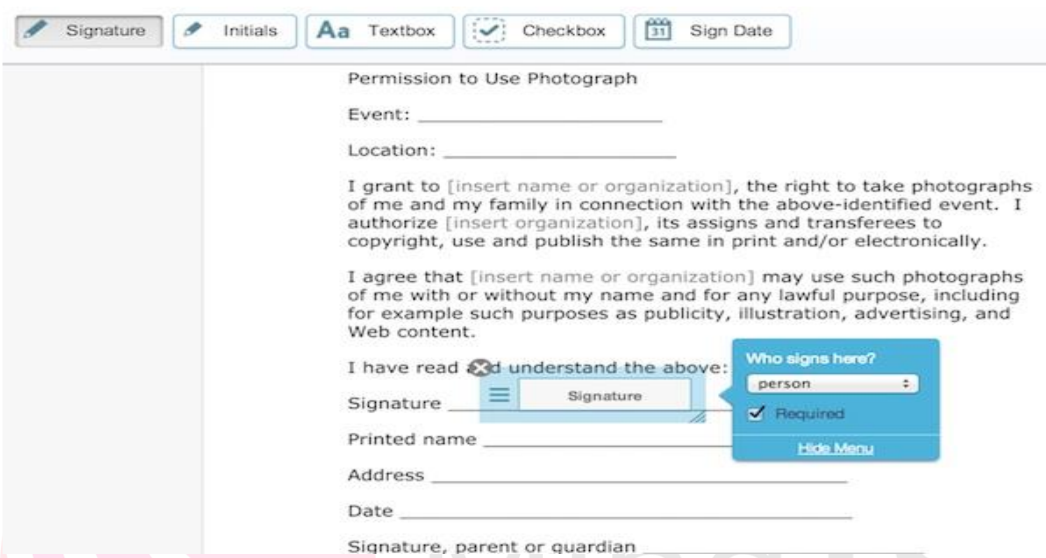


  - ✓ On clicking upload file option user can select the file of any format from the computer, and upload it.

  - ✓ After uploading, the app converts the document in user friendly format, where he can put signature, date, texts and other details.

  - ✓ Then under the title “Who needs to sign” the user will find boxes where he needs fill the signer's name and email address. If user needs to add more signers, look below the boxes and click the + Signer button. User can also copy other email addresses to the form from the + CC button.

- ✓ Now click on"PREPARE DOCS FOR SIGNING". When you do this, a popup window will appear that contains your document. Look towards the top of this document at the row of buttons labeled signature, initials, textbox, checkbox, and sign date. Drag-and-drop each of these boxes to their respective place on your document. For instance, drag a signature field to the signature area on your document.If you have multiple signers, make sure to select who signs and fills out what information from the drop-down menu above the respective box.

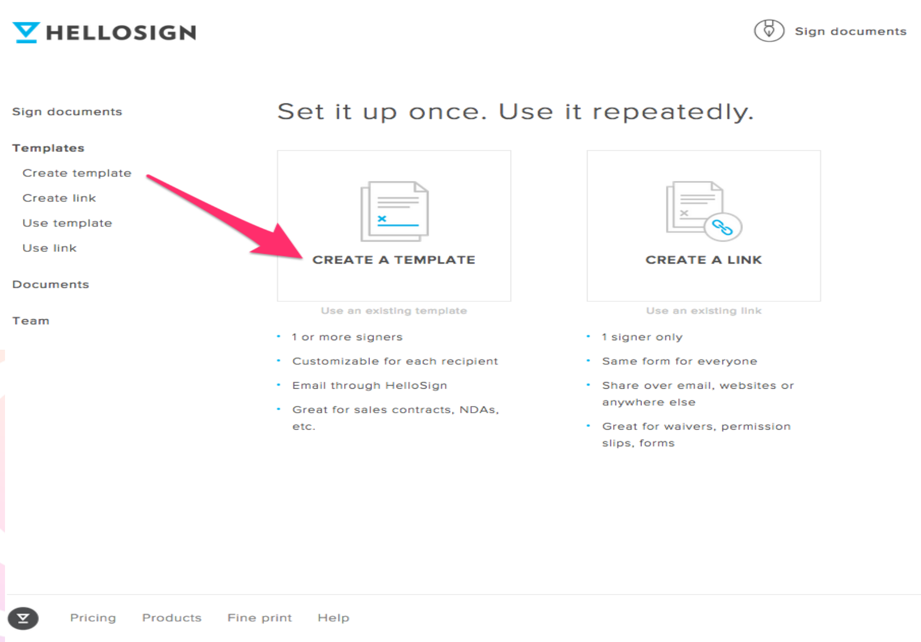


- ✓ On the bottom of the page under the Add document title and custom message header, the user can write a message for signees and name the document before sending it to signees.

TEMPLATES

CREATE TEMPLATE-

1. To create a template, user should login HelloSign account and click on "Create template" in the left side navigation bar. Select the "Create Template" option on the left.



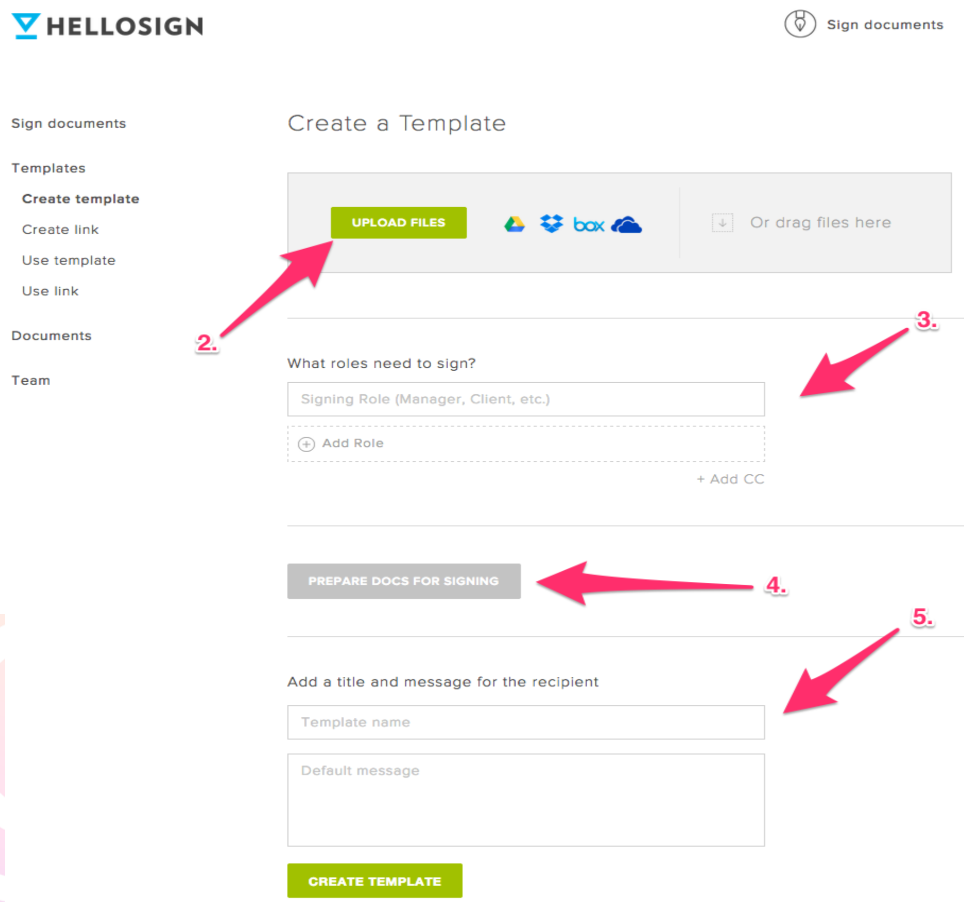
2. Upload the document user will be using as a template.

3. Create the necessary signing roles if only one person needs to sign, then that is one role. Click +Role to add additional signers.

4. Click the prepare docs for signing button. This will open a preview window user can use to set up the form. Select the text, date, signature, or check box field at the top of the screen, and then click wherever you want that field to be placed on the form. You can drag the field around to reposition it.

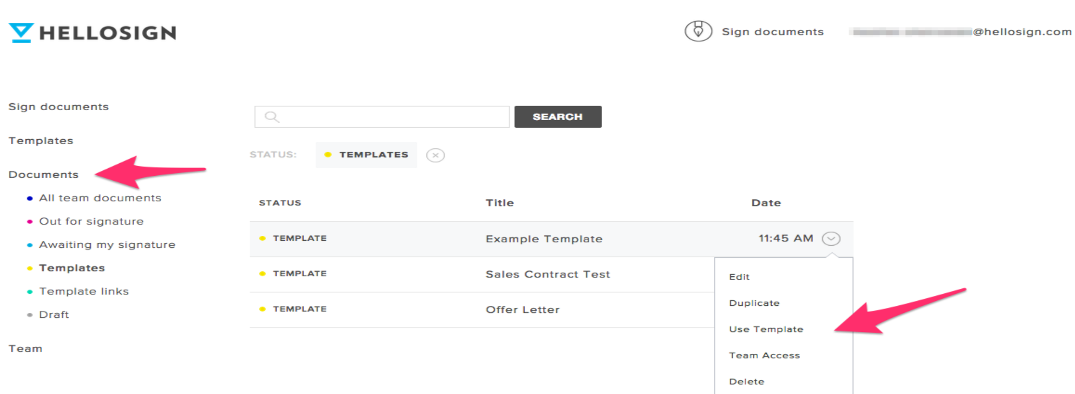
If you need to fill in certain variable fields each time you use the template, user can do so with our merge fields feature.

5. Once user has finished, he can save the document, give it a title and click "Create Template."



Whenever you need to use the template, click on "Documents" in the left navigation and click

To see all the templates, click the down arrow to the right of the template you want to use, and select "Use template."



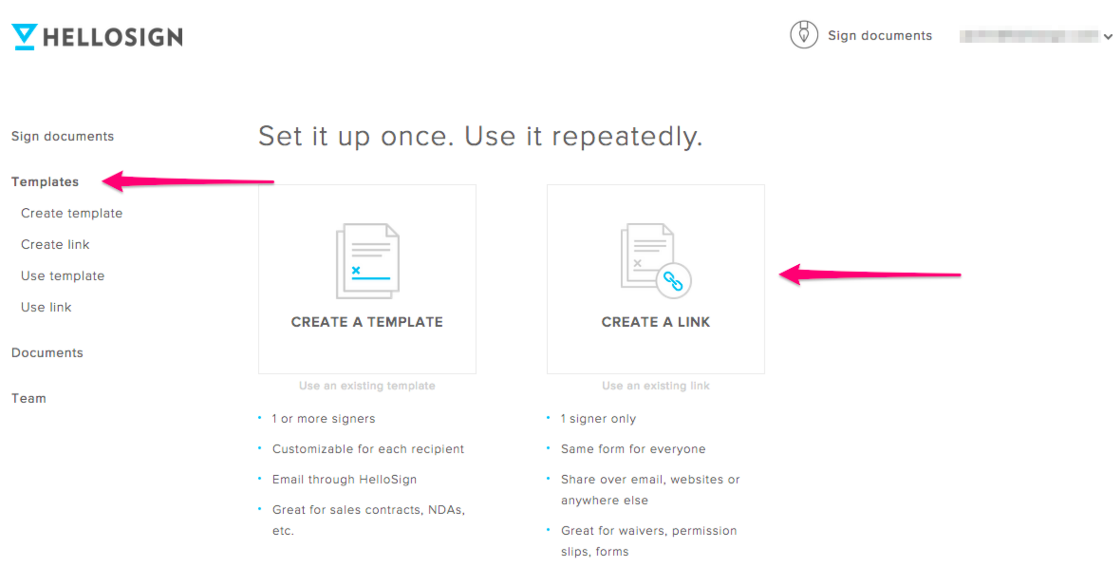
CREATE A TEMPLATE LINK-

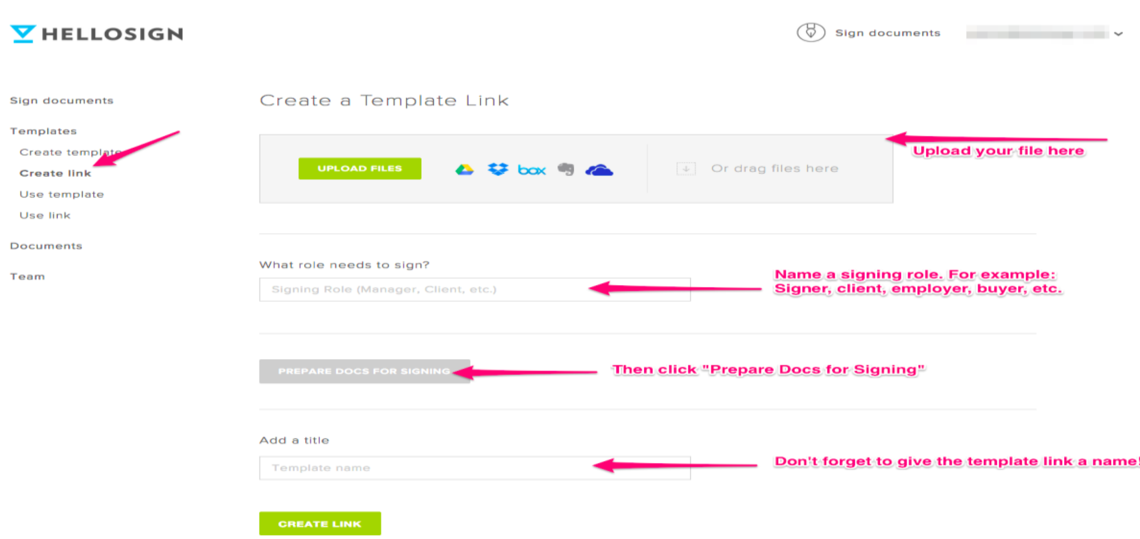
1. To create a template link, user should login HelloSign account and click on "Template" in the left side navigation bar.

2. Select "Create a link."

3. Upload the document you want to use and name the signing role for the link. (If there is more than one signer for the document you need to create a templateinstead).

4. Click the prepare docs for signing button. This will open a preview window you can use to complete the form.





Use the text field to create space for the signer to fill out. User can either drag and drop the button, or simply click the text box option and click wherever you want the text box to go on the document. From there user can drag it around. The same goes for the signature, date, and check box options.

Once user has finished, he can save the document, and give it a title. He'll get a URL you can use to email out to whoever needs it, or post it on your website. Every time a person clicks on the link, he or she will be guided through the document.

If user needs to get the link again, click on "Documents" in the left bar and use the 'filter by' drop down menu to search for Templates. Select the template link you want to use. The side bar will pop out to the right, and the link will be listed there for you to copy.

DOCUMENTS-

1. On clicking documents, we get all the information about the previously sent documents, title of document, date and its status.

- Out for signature
- Awaiting my signature
- Completed
- Templates
- Template links
- Draft

2. Red icon stands for “Out of signature”, blue icon stands for “awaiting my signature”, green icon stands for “completed”, yellow icon for “templates”, dark green icon for “ template links” and grey icon for Draft.

Status	Title	Date
OUT FOR SIGNATURE	CLOUD GMP-Ba...ction Record	May 8
COMPLETED	CLOUD GMP-Ba...ction Record	May 8

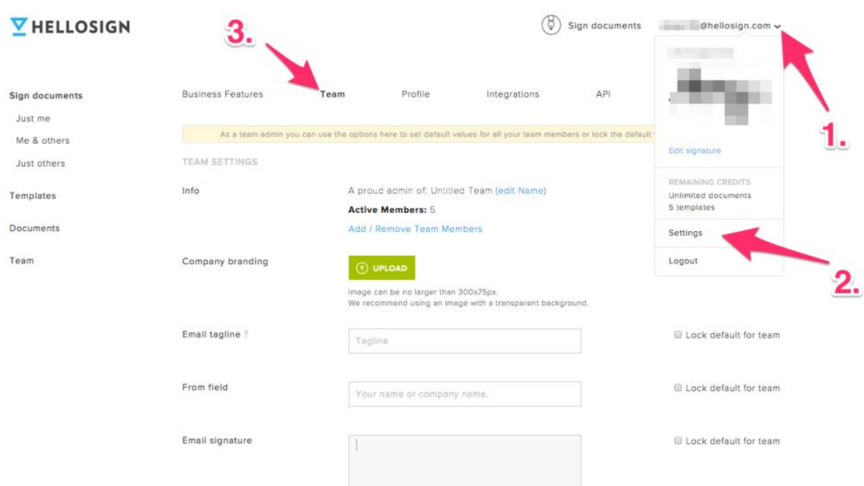
TEAM-

If you're a Business team admin, user is able to control and change the settings for your team as a whole from the "Settings" page. Here's how:

1. Click on your email address in the upper right corner.

2. Click on Settings in the menu that appears.

3. Click on the "Team" tab at the top of the page:

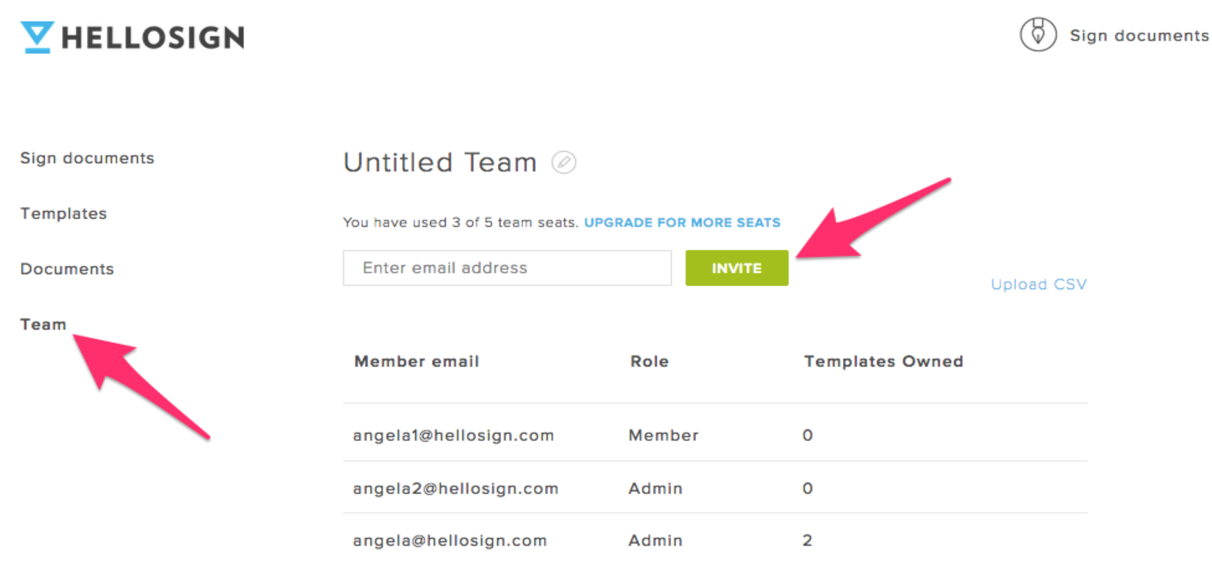


Here user will see where you can control and change how your HelloSign emails are branded.

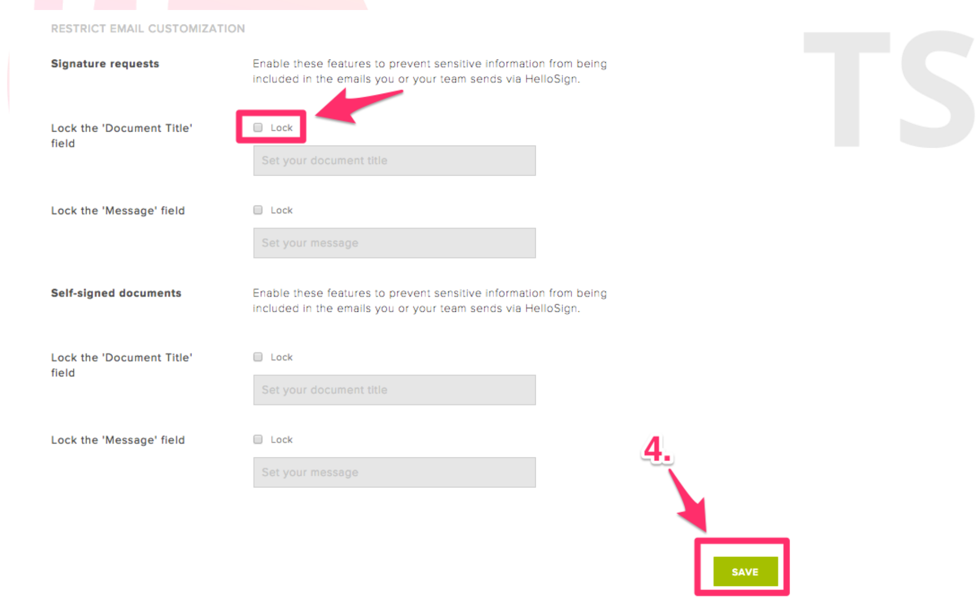
User can:

- Customize the "from" field to show your business's name instead of HelloSign
- Append a salutation in the footer of all emails sent
- Upload a logo to be placed on emails and on signature requests
- Change your industry
- Change the company name
- Change the date format
- Change the time zone
- Enable signature reminders

As the team admin, you are also able to lock these features so that you are the only person who is able to change them. If they are unlocked, any member of your team will be able to make changes as well.



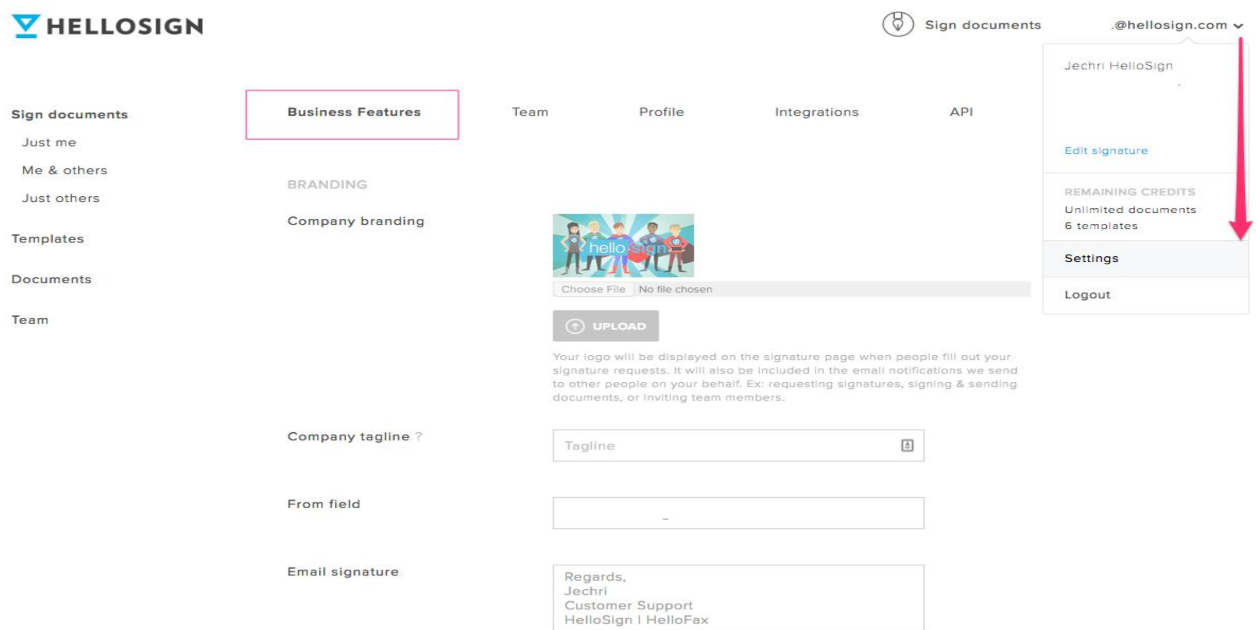
Once user has made the adjustments or locked any features, make sure he scrolls down and click "save" at the bottom of the page to save your changes:



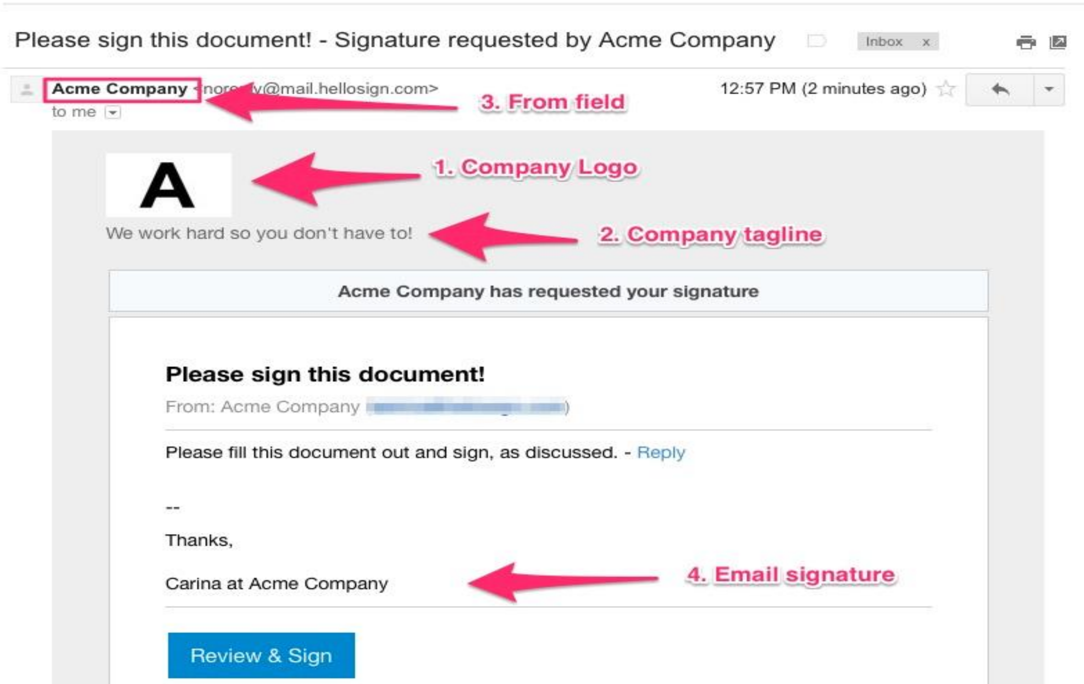
SETTINGS- In Business features, we get the option of branding.

Branding signature requests to customers and clients-

There are five ways hellosign business senders may brand their accounts. The team admin may control team branding by going to Settings > Business Features.



The branding options are:
1. Upload a company logo.(Image can be no larger than 300x75px.)
2. Create an e-mail tagline
3. Create a "from" field.
4. Create an e-mail signature to appear in all signature request e-mails
5. Take signers to a custom url after they've completed a signature request



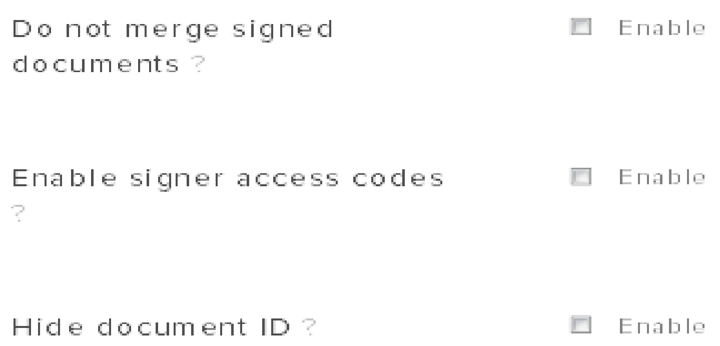
PREFERENCES-

Here user can select the form of signature is accepted. User can accept signature which would be typed, drawn, uploaded or with the help of smartphone.

Accepted signature types

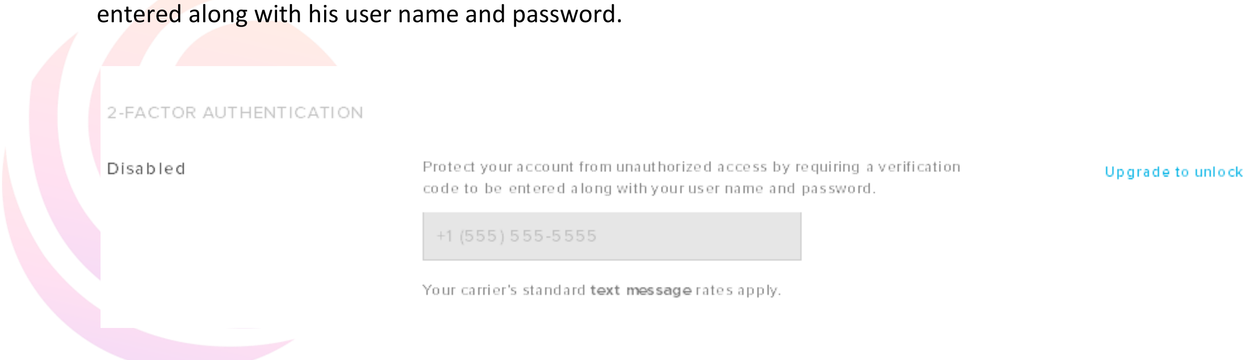
- [x] Drawn
- [x] Typed
- [x] Uploaded
- [x] Smartphone

Similarly, user has the option to enable the option of “not to merge signed documents”, “enable signer access codes”, “hide document id”.



- FACTOR AUTHENTICATION

  The user can protect his account from unauthorized access by requiring a verification code to be entered along with his user name and password.



- IN-PERSON SIGNING- This feature enables the people to sign users template links on the spot using a tablet.

PRICING (Membership)-. We will be offering various membership plans to users (Membership plans need to be updated by client) below are the Membership Plans offered by Hello Sign.

Hello Sign Membership Plans:

A wide range of services is available under the plan which the user subscribes. HelloSign has a limited free plan. With the free plan, user is able to make 3 document transactions per month, whether signing a document or requesting someone else's signature on a document. Users 3 free document transactions renews every 30 days from the day you signed up. Under PRO, BUSINESS, ENTERPRICE plan the user is able to get additional features as mentioned below.(Client needs to mention about his pricing policy as well as the services he wants to offer hi customers under various plans.)

Hellosigns plans and services

FREE	PRO	BUSINESS	ENTERPRISE
		1 - 5 SENDERS	
$0	$15 per month	$50 per month	Looking for more?
You are currently on this plan	GET STARTED 30-day free trial	GET STARTED 30-day free trial	CONTACT US (415) 766-0273 FREE
3 Documents per month	UNLIMITED Documents per month	UNLIMITED Documents per month	
1 Sender	1 Sender	5 Senders Included	
0 Templates	1 Template	5 Templates	
Integrated with Google Drive	Integrated with Google Drive	Integrated with Google Drive	
Status notifications	Status notifications	Status notifications	
Audit Trail	Audit Trail	Audit Trail	
	Data validation	Data validation	
		Branding	
		In-person signing	
		Signer access code	
		Team transaction overview	

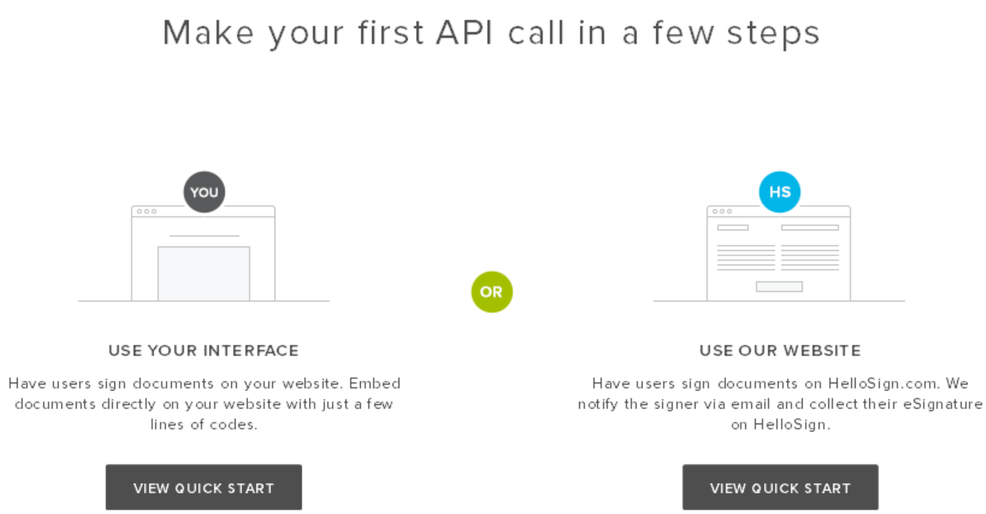
API-

(https://www.hellosign.com/api/documentation#QuickStart)

API Documentation/ tutorial will be provided to users on the website which will be managed by Admin

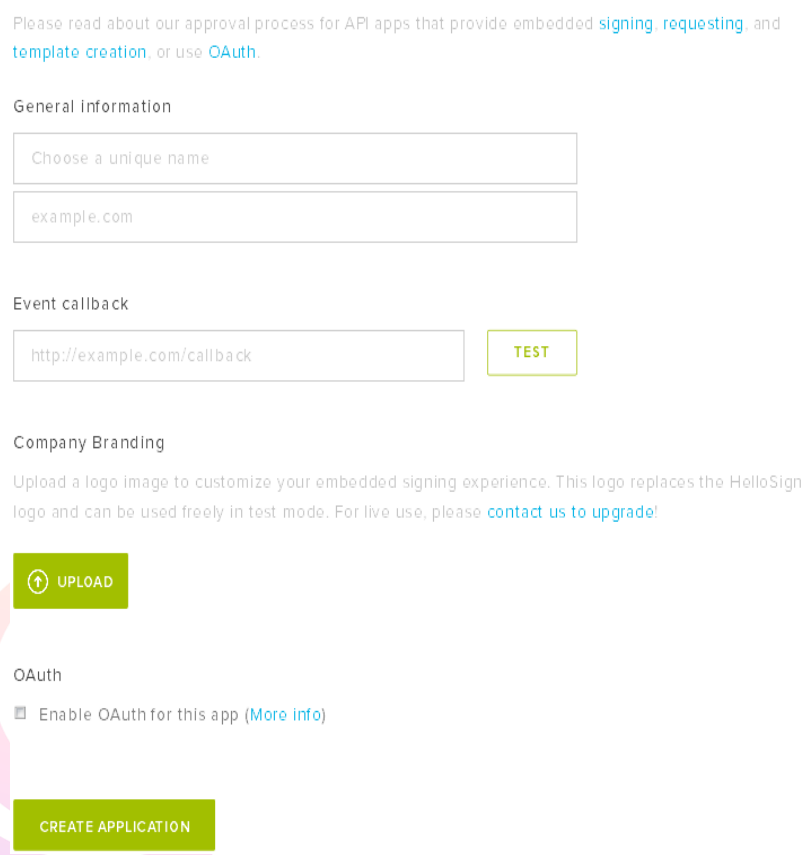
User can make his API call by two ways-

1. USE YOUR INTERFACE.
2. USE OUR WEBSITE.



1. **USING YOUR INTERFACE**- On clicking using your interface a unique API key would be generated. There would be an option of "create a test app" and on clicking on it.

CREATE A NEW API APP:



Testing

API Dashboard-With the help of API Dashboard the user is enabled to see his API requests and responses, view the status of your signature requests, and inspect his callbacks for easy troubleshooting and debugging

Authentication- 1. API primarily uses basic authentication to identify users when processing requests. Authentication data is securely encrypted using SSL.

- OAuth- HELLOSIGN support OAuth (2.0 only) as a way of authenticating.
- Permissions- If user has a hellosign team, the user must be designated as admin or developer in order to access API and view or set team roles on team page.

How we charge- The creation of live signature requests is not free and requires a paid API plan. The API will return HTTP 402 if such requests are made without a proper plan.

We understand that you may want to test the API before paying for it. The only API endpoints that are restricted to paid plans are the ones used to send signature requests. However, they can still be used without signing up for a paid plan by using our test mode.

Rate limits- By default, you can make up to **2000** requests per hour for standard API requests, and **500** requests per hour for higher tier API requests. In test mode, you can do 50 requests per hour. Exceptions can be made for customers with higher volumes.

Api Reference (https://www.hellosign.com/api/documentation#QuickStart)

Objects and Methods-

1. Account
2. Signature Request
3. Reusable Form
4. Template
5. Team
6. Unclaimed Draft
7. Embedded
8. Event

Callbacks There are two kinds of callbacks that can be defined:

1. Account Callbacks- All signature request events that involve your account are reported to this URL. This includes events such as receiving a signature request, or when a signature request you've sent is signed by someone. Events are reported to the account callback URL for either requests originating on hellosign.com, or on a partner site, via an embedded flow.
2. App Callbacks- This type of callback URL is set up at the API app level. API apps are used to identify a partner integration and configure embedded flows (embeddedsigning, embedded requesting, and embeddedtemplates) and OAuthproviders.

Call Back Response Format :-

The default format for messages sent to your callbacks is a multipart ('form/multipart') POST request, with the message details in a POST field named 'json'.

The Java SDK includes an event parsing class, and you can access this field in the $_POST array on PHP. For other platforms, either a library function, such as cgi.parse_multipart (for python) or ActionDispatch::Http::UploadedFile for Rails, or third party options, such as multer for node, may be useful.

Warnings and Errors- If an error or warning occurs; hellosign will return either an error or warning object in the JSON response.

Constants-(https://www.hellosign.com/api/reference#Objects)

Field types- These are the options you can specify for the "type" field.
Error names-List and details of possible errors that can be returned.
Warning names- List and details of possible warnings that can be returned.
Event names- Here are the list of possible events that can be sent to your callbacks.

Signature status codes- Callback events come along with a SignatureRequest object, under which you can find the list of associated Signatures. Each of these Signature object has a status_code field that describes its current state.
Data validation types-Text fields accept an optional validation type, which must be met before the user can submit the signed document. This value can be specified in form fields, text tags, or on the web. These are the options you can specify for validation type.

Search- Searches can include queries on specific fields when hitting the "/v3/signature_request/list" or "/v3/template/list" endpoints.

Search fields-Search terms that are not passed with a specific field will be matched against: to, from, title, subject, message, metadata, and filename. Queries on fields can be combined using AND and OR operators (operators are case sensitive). Dates may be passed in as ranges. Exclusive ranges are indicated with curly brackets {min TO max} and inclusive ranges with square brackets [min TO max]. Leading wildcards (*) are not valid queries. The table below lists all possible fields that can be queried.

White Labeling- White labeling is a term for branding a product that you have purchased. Our white labeling allows you to personalize the logo, color scheme, and legal text of the signer page, making our product blend seamlessly into yours.

Before you can do anything with white labeling, you will need to create an app. You must have a Sapphire API plan (or higher) to use white labeling in production mode. Otherwise, you can only use it in test mode. All updates made to white labeling options by users with a minimum Sapphire API plan are immediately active in production. We recommend creating a test app for previewing white labeling changes.

White labeling is available exclusively on the new signer interface. For users with older accounts, please see the embeddedsigningwalkthrough for more info on setting the uxVersion.

The white labeling options are passed in the APIappandupdateAPIappendpoints.

Custom colors- All customizable color elements require valid six-character hexadecimal codes for values. All elements are optional, and any element that you choose not to customize will be set to its default value. If you choose not to customize the value for an element that has a hover state, its hover state will inherit from its active state.

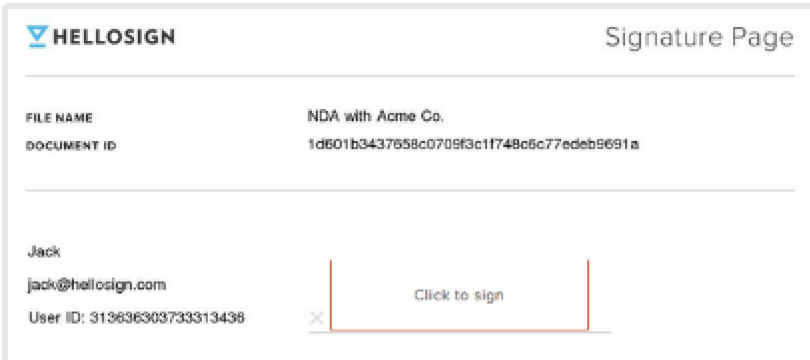
Legal version-The legal version element is related to our terms of service.
Error types- Validation errors are JSON-encoded and returned as a value on *error_msg* (for more info on errors, see the errors and warnings guide). An error includes the error type and the invalid element name or names.

WALKTHROUGH (https://www.hellosign.com/api/signatureRequestWalkthrough)

Sending Signature Requests

There are several ways to send a signature request using the HelloSign API. The method you may want to use will depend on your use case. Take a look at the different options we offer to decide which scenario fits your needs.

USING FILES- This is the simplest way to create a signature request, attach the file(s) along with signer details and you're all set. A signature page will be appended at the end of your documents with a signature field for each signer involved.



USING TEMPLATE-Templates allow you to specify form fields (text, check boxes, signatures, etc...) that the signers should fill. But it also offers the possibility to define "custom fields" (see picture) that will be filled when the request gets sent via the API and will be shown to the signers.

USING FORM FIELDS- If you need to send documents with changing content but the same form fields, one can use form fields rather than templates.

Sending Signature Requests

There are several ways to send a signature request using the HelloSign API. The method you may want to use will depend on your use case. Take a look at the different options we offer to decide which scenario fits your needs.

Scenarios

How do you plan on creating your signature requests?

Using files

You provide the files to create the request and we add a signature page for the signers to sign.

Using templates

Setup a template ahead of time on hellosign.com and reuse them with custom data it when sending signature requests.

Using files and form fields

You provide the files and tell us exactly where form fields go on them. No signature page is added in this case.

<u>Finding out when documents have been signed</u> -

There are 2 ways get the status of a document. The preferred method is to provide a callback url which we'll post events to. The option to poll the API is also available but it's impractical in most situations since signers may take a while to view and sign your documents.

Web hooks (preferred) - Read our documentation about events to get a better understanding of how web hooks work with your account and apps.

Polling- (Warning)If you poll, any frequency you choose has disadvantages, either having to make thousands of requests per signature request over a day or two or making fewer requests and having more time pass between the signature request completion and your system finding out about it. Web hooks solve both of these problems.

Retrieving documents- At any given time during the signature request process, you can retrieve the associated document in its current state via the API. You can either download a single PDF document or separate pages in a zip file. Either option will contain all the signatures and information that signers have added so far.

Using templates with the API- (https://www.hellosign.com/api/templatesAndApiWalkthrough)
Template setup

The following steps walk you through the process of creating a template using the web interface.

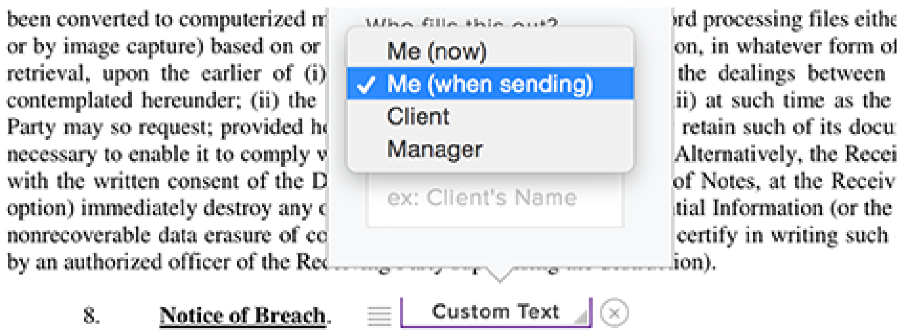
On HelloSign , create a template. Upload a document and specify the roles for each signer, for example 'Client' and 'Me'.

Click "Prepare Documents" and add fields to be filled out by each role by dragging them from the top bar down to the document.

Add fields for any text or checkmark annotations you'll want to make at the time of sending.

In the field pop-up menu under "Who fills this out?", select "Me (when sending)." This designates the field as a custom field that will be populated by your API request when you use the template to send a signature request.

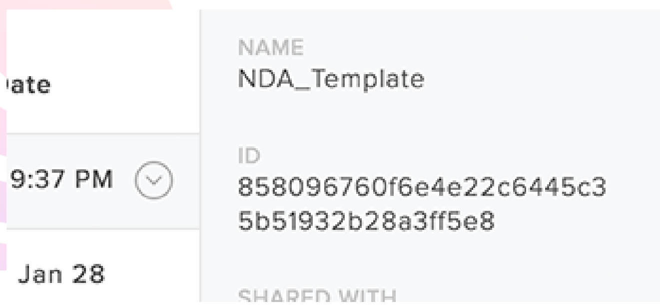
You must also specify a Field Label that you will use to reference the custom field in your API request.



Finding a template ID via hellosign.com

You will need to find a template's id in order to be able to use it with the API.

Find the template on the documents page and select it. The id is listed in the slide out info panel.



Finding a template ID via the API

You can retrieve a paged list of your templates by making a GET call to template/list. Iterate through this list, requesting additional pages if necessary, to find the template that has a matching title and/or signer_roles. You can then retrieve the ID from this template object's template_id field.

Using a Template to Send a Request

You can send a signature request based on a template in your account by making a POST request to signature_request/send_with_template and including the **template_id** parameter. You will also be required to provide a unique signer name and email address for each signer role defined in the template.

Using a Template with Custom Fields

When sending a signature request based on a template, you can also pass in custom fields as an optional parameter. The following takes the POST request in the previous section and includes custom field data. The signer role 'Client' is allowed to edit the value start_date.

Using Multiple Templates in a Request

You can send a signature request based on multiple templates in your account by making a POST request to signature_request/send_with_template and including the necessary **template_ids[%i%]** parameters. You will be required to provide a unique signer name and email address for each signer role in each template. One exception to sending a request with multiple templates is that you cannot use more than one template with ordered signers.

Using Long Text in Custom Fields

The text for a custom field should fit into the space available. If it does not fit it will overrun the box and a warning will be issued. Alternatively, if the signer is assigned editor privilleges to the custom field and the text overruns the border, an error will be returned. The exact length of text that will fit in the space depends on the nature of that text. If the text uses words with wide letters, or uses words that are long and do not wrap efficiently, few words will fit. If the text instead uses small skinny words, lots of words will fit.

Some guidance is given to help you choose an appropriate text length. Refer to "Finding a template ID with the API", the response object contains **avg_text_length**, which contains the expected number of characters that will fit on each line of the custom field, and how many lines will fit in the custom field. You can use this to choose the best text length. If you exceed the number of characters that fit when sending a template, the warning message will repeat this suggest length information.

It is possible to send fewer than the suggested number of characters, and to still get the warning message. This is because you may be sending unusually wide words, or words that wrap into the available space inefficiently

Embedded Signing- (https://www.hellosign.com/api/embeddedSigningWalkthrough)

Preliminary

Before you start doing anything, make sure the following steps have been done:

1. Create an API app (login required)

Server Side- The very first step is to create an embedded signature request from your backend using our API and then fetch the associated url that will be used in the iFrame.

Client Side- We provide a library that takes care of building and displaying the iFrame on your page for you. It should only take you a few lines of JavaScript code to make this functional.

App Approval Process for Embedded Signing

1. **Review** our user identification requirements before adding Embedded Signing to your website:

- Your users must be authenticated on your website before they may sign documents.
- We also require that your users have previously confirmed ownership of the email address used to identify them to HelloSign. In practice, verifying email ownership means that when a user signs up for an account on your website, they should receive an email containing a unique link back to your site. When followed, your application can record the action, thus verifying the user's ownership of the email address.

2. **Email** us at apisupport@hellosign.com to schedule a certification review once your integration is complete and ready for production use.

3. **Demonstrate** your completed integration via screenshare (using a Google Hangout or Skype) by walking us through user account creation, email confirmation, and login on your website. We will also review your HelloSign integration to make sure the user can sign documents without any errors. A review typically takes less than 30 minutes.

Additional notes –

Domain restriction- The iFrame checks that the parent window belongs to the domain associated with your app. If it does not, the HelloSign page will not be displayed.
If you wish to disable the domain restriction to make development easier, you may set skipDomainVerification to true when calling HelloSign.open .

Redirection-

If a redirect url is specified, the iFrame will redirect users to it after they sign. Data from the EVENT_SIGNED event will be appended to the url query string.

Signature links

Signature URLs are only valid for 60 minutes after embedded/sign_url has been called and expire as soon as they're accessed.

Text Tags

The embedded functionality can be used in conjunction with HelloSign Text Tags.

Events

Here is the detail of all possible events the iFrame could send to the opening window.

Embedded Requesting-(https://www.hellosign.com/api/embeddedRequestingWalkthrough)

Preliminary

Before you start doing anything, make sure the following steps have been done:

1. Create an API app (login required)

   Server Side
   The very first step is to create a draft signature request from your backend using our API. You must decide what you want to happen after the signature request is sent by your user. By default, HelloSign will send an email to the signers and ask them to come to hellosign.com to sign and complete the document(s).

Client Side-

We provide a library that takes care of building and displaying the iFrame on your page for you. It should only take you a few lines of JavaScript code to make this functional.

App Approval Process for Embedded Requesting-

Below we have included the process followed by Hello Sign. We need to know whether to implement same process or define any other process

- ***Hello Sign Process:***

In order to ensure that your integration adheres to eSignature regulations and best practices, we require a quick certification review.

NOTE You will still be able to use your app in test mode until it gets approved. Only live activity requires approval.

1. Review our user identification requirements before adding Embedded Requesting to your website:

- Your users must be authenticated on your website before they may request signatures.
- We also require that your users have previously confirmed ownership of the email address used to identify them to HelloSign. In practice, verifying email ownership means that when a user signs up for an account on your website, they should receive an email containing a unique link back to your site. When followed, your application can record the action, thus verifying the user's ownership of the email address.

2. Email us at apisupport@hellosign.com to schedule a certification review once your integration is complete and ready for production use.

3. Demonstrate your completed integration via screenshare (using a Google Hangout or Skype) by walking us through user account creation, email confirmation, and login on your website. We will also review your HelloSign integration to make sure the user can request signatures without any errors. A review typically takes less than 30 minutes.

That's it! We'll immediately approve your app (or make suggestions) after the review.

App Approval Process for Embedded Requesting

In order to ensure that your integration adheres to eSignature regulations and best practices, we require a quick certification review.

Additional notes

Domain restriction The iFrame checks that the parent window belongs to the domain associated with your app. If it does not, the HelloSign page will not be displayed.

If you wish to disable the domain restriction to make development easier, you may set skipDomainVerification to true when calling HelloSign.open .

*Redirection-*If a redirect url is specified, the iFrame will redirect users to it after they send the document(s). Data from the EVENT_SENT event will be appended to the url query string.

Text Tags- The embedded functionality can be used in conjunction with HelloSign Text Tags.

Events- Here is the detail of all possible events the iFrame could send to the opening window

Embedded Templates-(https://www.hellosign.com/api/embeddedTemplatesWalkthrough)

Embedded Templates

Embedded templates allow your users to create and edit templates on your site in an iFrame. Templates are used to share frequently-used documents and send them for signatures.

Preliminary

Before you start doing anything, make sure the following steps have been done:

1. Create an API app (login required)

   Creating a template
   Allowing your site users to create a template involves first creating a template draft. This draft will be displayed on your site in an embedded iFrame, allowing the user to edit and finalize the template.

Pre-defined merge fields-By creating a template draft with merge fields, you are essentially telling your users, "The app can supply this information for you when you use the completed template to send a signature request."

Editing a template

Once a template has been created, you can also open an embedded iFrame that allows your site users to edit it.

Previewing a signature request-

When your site user needs to send a signature request based on a template, you will need to create an embedded signature request draft. This draft can be opened in an embedded iFrame to allow the user to preview the request prior to sending. Creating the signature request draft for previewing is also a perfect opportunity to populate any pre-defined merge fields that might have been added by the user during the template creation process.

App Approval Process for Embedded Templates

In order to ensure that your integration adheres to eSignature regulations and best practices, we require a quick certification review.

NOTE You will still be able to use your app in test mode until it gets approved. Only live activity requires approval.

1. **Review** our user identification requirements before adding Embedded Templates to your website:

- Your users must be authenticated on your website before they may create templates.
- We also require that your users have previously confirmed ownership of the email address used to identify them to HelloSign. In practice, verifying email ownership means that when a user signs up for an account on your website, they should receive an email containing a unique link back to your site. When followed, your application can record the action, thus verifying the user's ownership of the email address.

2. **Email** us at apisupport@hellosign.com to schedule a certification review once your integration is complete and ready for production use.

3. **Demonstrate** your completed integration via screenshare (using a Google Hangout or Skype) by walking us through user account creation, email confirmation, and login on your website. We will also review your HelloSign integration to make sure the user can create templates without any errors. A review typically takes less than 30 minutes.

That's it! We'll immediately approve your app (or make suggestions) after the review.

Additional notes

Domain restriction- The iFrame checks that the parent window belongs to the domain associated with your app. If it does not, the HelloSign page will not be displayed.

If you wish to disable the domain restriction to make development easier, you may set skipDomainVerification to true when calling HelloSign.open .

Redirection- If a redirect url is specified, the iFrame will redirect users to it after they sign. Data from the EVENT_SIGNED event will be appended to the url query string.

Signature links-Signature URLs are only valid for 60 minutes after embedded/sign_url has been called and expire as soon as they're accessed.

Text Tags-The embedded functionality can be used in conjunction with HelloSign Text Tags

*Events-*Here is the detail of all possible events the iFrame could send to the opening window.

Events and Callbacks- (https://www.hellosign.com/api/eventsAndCallbacksWalkthrough)

Events will be POSTed to your callback URLs. There are two kinds of callbacks that can be defined:

Account Callbacks

This type of callback URL can be set up at the account level (either by setting it through an API request to "/v3/account" or on the Account Settings page on hellosign.com).

*App Callbacks-*This type of callback URL is set up at the API app level. API apps are used to identify a partner integration and configure embedded flows (embedded signing, embedded requesting, and embedded templates) and OAuth providers.

Callback Response Format

The default format for messages sent to your callbacks is a multipart ('form/multipart') POST request, with the message details in a POST field named 'json'.

Responding to callbacks

Your endpoint will need to return a 200 HTTP code and a response body containing the following text: Hello API Event Received. Otherwise, the callback will be considered a failure and will be retried later.

The Event Object

Events posted to your callback url will be formatted as a JSON string contained in the json POST param. The complete detail for the event object can be found in the event section of the API reference page.

Callback HTTP headers

Event Hash Verification

Each event object in the callbacks contains an event_hash to help you verify that it's really coming from HelloSign. The event_hash is generated with the HMAC algorithm using your API Key as a key and SHA256 digest mode.

Failure and Retries

If your callback url is not reachable or returns a non-successful response, we will retry POSTing the event up to 6 times, with each retry interval being exponentially longer than the previous one.

Please note that our requests will timeout after 30 seconds, so callbacks will fail if your server takes longer than that to respond. The retry pattern is described below, an alert email will be sent to you after POSTing has failed several times.

Getting Started with OAuth 2.0(https://www.hellosign.com/api/oauthWalkthrough)

App Setup-

Create an app- Your app will need to be approved before anyone but you can authorize it. When you feel ready to go live, email us at apisupport@hellosign.com to request approval. You will be notified via email when the app is approved. Until then, you will be able to authorize it yourself and use it in test mode

App attributes

Available scopes

App Approval Process-

In order to ensure that your integration adheres to eSignature regulations and best practices, we require all OAuth apps to email us at apisupport@hellosign.com when ready for approval

Authorization

A user needs to authorize your app before you can act on his behalf. Once they do, your app will be granted an **access token** which can be used for authentication when making requests on behalf of your users.

Checking for an existing HelloSign account

There are two major authorization scenarios, one for existing users and one for new ones. This is why we provide a way for you to check if a user exists.

Retrieving the OAuth token

Authorization from new users

If the user does not already have a HelloSign account, you can still send them to the authorization page and they will have to signup at that point. Preferably, you will want to avoid this. Instead you should create an account and get the authorization data at the same time.

Token Refresh

Access tokens are only valid for a given period of time (typically one hour) for security reasons. Whenever acquiring an new access token its TTL is also given (see expires_in), along with a refresh token that can be used to acquire a new access token after the current one has expired.

Making API Calls-

Using Text Tags Field Parsing-(https://www.hellosign.com/api/textTagsWalkthrough)

Matching Signer API Parameters-

Defining Tags as Variables-

Tag Removal

There is a parameter than can enable automatic tag removal called "hide_text_tags". While this may seem convenient, we actually don't recommend it. The reason is that automatic tag removal can be an inexact science and unwanted clipping may occur.

Usage Options

Sizing and Substitution

By default, the new field has the same width and height as the original tag text. The height is always fixed to the text height that HelloSign offers the signer, but you can change the width of the tag.

Labels and Unique IDs

Possible Problems

Allowed Characters

Tag elements can be made of alpha-numeric characters, and the underscore "_" character. Tag types and signer numbers are case insensitive, but tag variables are case sensitive and case is preserved in the label.

Tag detection

Tags are detected by text value, not by OCR, so the documents supplied must contain literal text for tags. PDF files are recommended, because the location of the tags is important. With other formats, such as plain text, the tag may end up in a different location than expected. The tag may even wrap from one line to the next, which will cause the tag to be ignored.

Specifying Signers

Signers must be part of a list without gaps, so a set of documents could not specify only "signer1" and "signer3". Every signer mentioned in a tag must be specified in the signature request. Every signer specified in the signature request must be in a tag. If the document refers to "signer1" then in the signature_request API request there must be a "signers[1][name]" and a "signers[1][email_address]" parameter set.

Tag Removal

We recommend writing your tags using a font that is not visible, such as using a white font on a white background. If you do NOT do this, the tags will be read and understood but the Text Tags will still be present on the document. You can have us attempt to hide the text by setting "hide_text_tags=1 However we don't recommend this as occasionally removing the tags on the backend can cause undesired clipping side-effects. For more information see the Tag Removal section.

Errors

If your signature request is not valid because of missing signer information or missing fields for signers to complete, there will not be an immediate error. Instead, you can be notified of the error as an event. The error event type will be signature_request_invalid. You can read more about this in our events and callbacks walkthrough.

Example Document

EXAMPLE-(https://www.hellosign.com/api/twilioExample)

Example: 2-Factor Authentication Using Twilio

In this example you will combine Twilio's API with HelloSign to add 2-factor authentication to a document. In addition to having identity verification from email, a secret code is sent via SMS to the recipient's phone. Without the secret code, no one will be able to sign the document.

Example: Send CLA After Github Pull Request(https://www.hellosign.com/api/githubExample)sss

Step 1: Get an OAuth token from Github

This step requires you to authenticate with GitHub. When we receivesssssss the response from GitHub, it will contain your access token and email address. This access token is used to create the pull request in step two.

We use a combination of Github API endpoints for OAuth, users and users/emails

Step 2: Generate a pull request

Your token: Complete step 1 first

To generate a pull request on our demo repository, run this code in your browser console or click below.

For simplicity, we already created a branch in the github repo hellofax/hellosign-cla-example that you will be pull requesting.

Step 3: Check your email
As soon as we receive the callback from Github that you created a pull request, we'll send you a CLA to sign. You'll also get an email from Github letting you know when we automatically close your request.

How it works

From our Github repo's settings, we set a callback to notify us whenever a pull request occurs. At that point it's only a matter of using our endpoint for signature requests via templates .

The CLA was preconfigured as a template. This is the best solution for a generic document that any number of people will sign. Optionally it could be customized on a per user basis to include extra prefilled data like someone's Github id.

The pull request callback only contains a GitHub id which was why we fetched the email address associated with it in Step 1. Alternatively you could use the account's public email address if it had been set.

Example: Embedded Photo Release-
(https://www.hellosign.com/api/embeddedSigningExample)

We will provide a sample demo script to end user for Trial. They can use this script on their website to test how the system works.

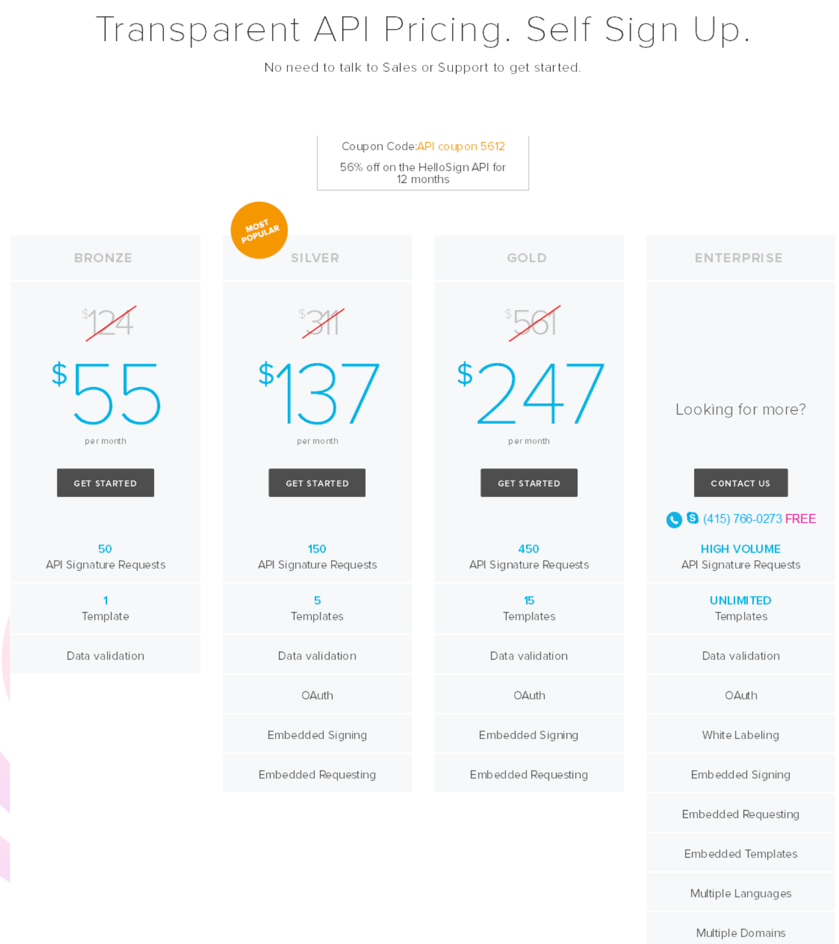
Libraries(https://www.hellosign.com/api/libraries#Official)

The HelloSign API is built to be simple and usable. We return JSON objects, use basic authentication, accept POST parameters and do whatever else we can to make your life easier.

Official HelloSign Libraries-

Once the APIs are developed we will provide users to download API code through this section

PRICING (Membership) - We will be offering various membership plans to users (Membership plans need to be updated by client) below is the Membership Plans offered by Hello Sign



OUT OF SCOPE

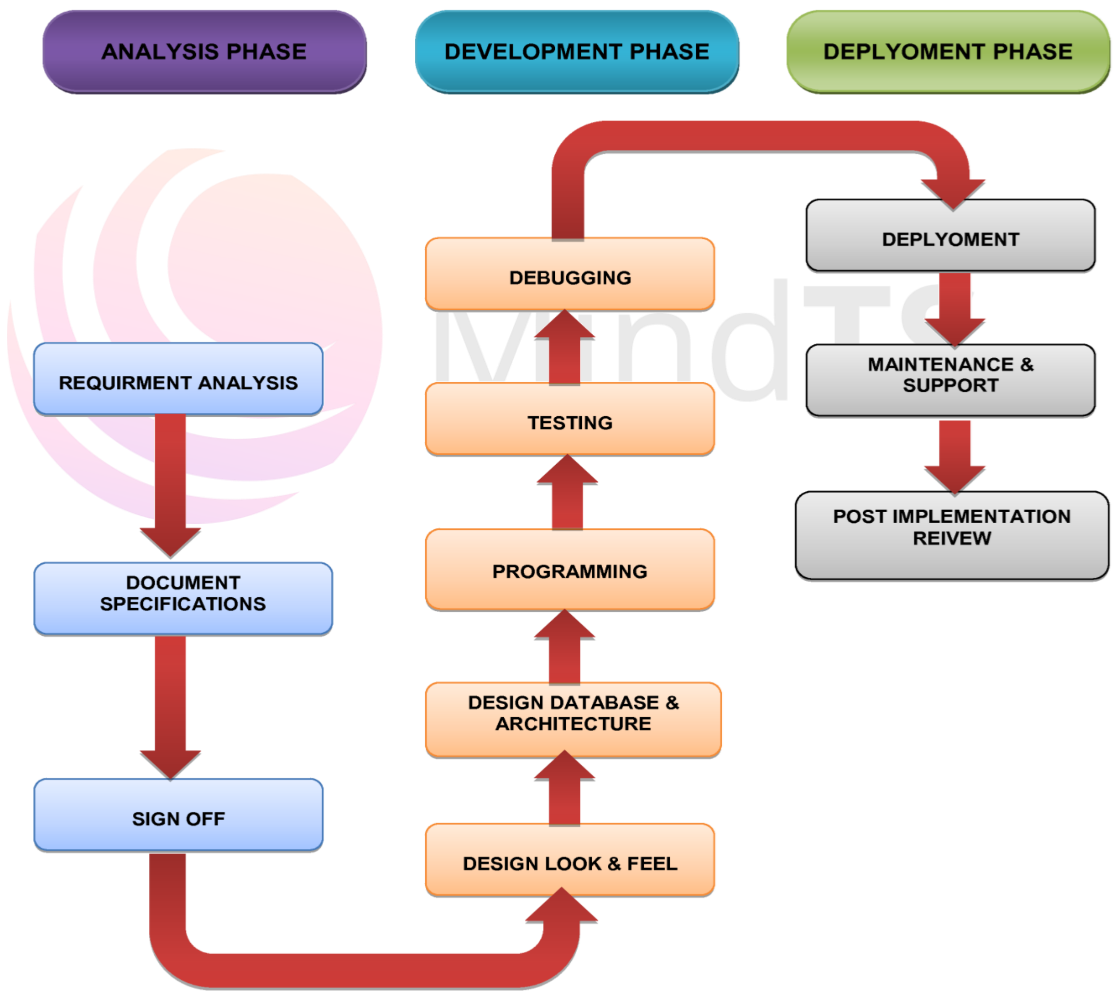
Hardware and 3rd party software costs are not included in the fees for this proposal.3rd Party API– This proposal assumes that Client will provide all the third party API. Like Payment Gateway, mass email service provider, chat API, SMS Gateway, and Pre done Plug-ins etc.

Cost of SSL and certificates is not included in the fees for this proposal. Data backup, disaster recovery requirements.

MINDTS WORK APPROACH

Note: This is our standard workflow, kindly note that we would provide the designs, deployment and maintenance

PROJECT DEVELOPMENT PROCESS



Support (______ month's free support) as per our commitment with the client.

Project Deliverables

Deliverables of this engagement document will include the following:

Pre-Production:

This step includes the generation of a basic application for framework for programming the website. This step will also include upon the finalization of the web directory structure. Graphic design options shall be presented to client based on the IDS and web design questionnaire for Client's approval.

Production:

The various activities in the Production / Creation of the website are given below:

Database Design in MySQL.

Integration of the Content Management System as specified by Client.

Programming of the Administration Area.

Programming of the User Area.

Creation and finalization of the graphics.

Creation of artwork for banners, icons, page hardware.

Integration:

Integration of the developed work together (Graphics and Programs)

Proof Reading:

This process will include checking for typographical errors induced in the production stage.

QA/ Testing:

QA will have layers of review that includes Peer Review and Review by the Lead Software Engineer. Apart from this the QA engineer will check quality of all the functionalities and links.

Delivery: The developed website will be delivered on to the designated server either using FTP or a File Upload application. Any server level permissions issues will also be handled in this phase.

Contact details

Head Office

01, Goshiba Majesty, Lancer circle, Ganngapur Road, D'souza Colony Nashik, Maharashtra 422005, India
Tel: +91 253 6494999

Corporate Office USA

610 Elm Street,
Suite 700
Mc Kinney, TX 75069
Tel: +1 (855) 964-6387