



THE LOANPRO API

REAL WORLD INTEGRATION

loanprosoftware.simnang.com

ALL RESOURCES

1. [Plunker](#)
2. [Articles](#)
3. [Simnang Forums](#)
4. If you are in a live workshop event, please raise your hand if you have a question!

PURPOSE OF THIS PRESENTATION

- All other presentations show how API calls work
- They use samples which show how *not* to integrate with the API
- This presentation talks about what to consider when writing integrations

PROBLEMS WITH THE API SAMPLES

- All depend on Angular
- All work only in the browser
- All expose API token to users
- Private forks of samples can be viewed by strangers if the "freeze" button is clicked

Conclusion: Don't use the API Samples for your integration!

IDEAL INTEGRATION

- API token stays secret (never leaves server or database)
- API token is centralized (not all over code)
- Can easily switch between production and staging
- Can add to existing systems

Due to the last point Simnang can never write an ideal integration for you

OPTION 1: USE AN SDK

SDKs exist for **PHP**, **Javascript (Node.js)**, and **C#**

PROS

- Don't send token to browsers
- Authorization is set in a single function call allowing centralization
- Can open issues on Github for bugs found

CONS

- Currently cannot easily switch between production and staging*
- Support limited to provided languages
- No "cool feature" or "it would be great if..." requests accepted

* Some SDKS can sort-of switch, some cannot

OPTION 2: FORK AN SDK

Instead of using vanilla SDKs, fork one and build your own

PROS

- Plenty of initial code to get you started (low cost and time to get going)
- Most of the benefits of the SDKs
- Can add cool features or fix bugs without communicating with Simnang

CONS

- Will need to merge bug fixes yourself
- Support limited to code in vanilla SDKs
- Still limited to provided languages

OPTION 3: DO IT YOURSELF

You can always write it yourself

PROS

- You design it, you define the features supported
- You can guarantee it works with existing systems
 - language, design, naming conventions
- You can add all the cool features you want

CONS

- It costs more (time and money)
- No support from Simnang
- May not have any of the ideal qualities

OPTION 4: FORGET INTEGRATION AND USE THE UI

You can always have humans use the UI

PROS

- Don't have to worry about maintaining servers
- Don't have to worry about API tokens
- Can easily switch between production and staging
- Can work into any existing workflow

CONS

- You have to work with humans (train, support, encourage)
- Your speed is limited by the speed of humans
- You have to pay them

WHAT TO USE?

Really small company

- just use the UI

If you can use a supported language

- Recommended to use a vanilla SDK
- Fork if SDK no longer matches needs

If using a supported language isn't an option

- Stuck with DIY

Recommended to use the UI where feasible

THINGS TO AVOID

- Never send the API token to a browser in any form
 - Encrypted or unencrypted, obscured or not
- Avoid having the API token appear more than once in your code
 - Use a DB or config file if needed
- Do not write more code than you need to
 - Over engineering
 - Writing too many "cool features"
 - Re-inventing the wheel
 - Writing code that will never be used

THINGS TO DO

- Create tests for your code
- Document your code
 - At least document how to communicate with LoanPro
- Use existing libraries, especially libraries that have been well-tested
- Define your needs and requirements before you start

QUESTIONS?

Feel free to ask in our [Forums](#)

Or, email us at support@simnang.com