where

VISION and PASSION

meet

TECHNOLOGY

ABOUT US

We are a team of **perfection seekers**, in **constant** search of ways to **improve** our work, processes, products, and services. **Everyone** at **znerg** shares the same **passion**, the same dream of building something meaningful and of great **value to our customers**. Ultimately, we want our customers to say **wow**!

SO
WHAT IS
THIS

znerg Architecture?

TURNING AN IDEA INTO A REALITY

YOU HAVE AN IDEA

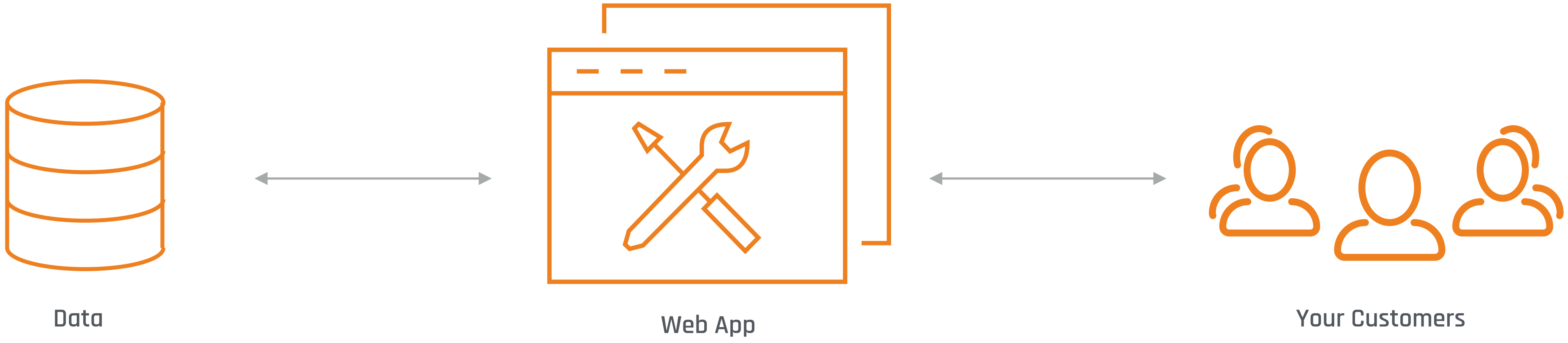
Generate value
Improve competitiveness
Increase market share

BY PROVIDING SERVICES with a WEB APPLICATION

YOUR GOALS

Excellent User Experience
High Quality
Rapid Development
Low-Cost Maintenance
Easy-to-Extend
Rich Features
Reasonable Price

CONVENTIONAL APPROACH



YOUR PLAN

Choose a framework
Use responsive design
Build a good team
Set your budget
Commit to a delivery date

So everything should work just fine right?

Not really

CONVENTIONAL APPROACH

What is the problem?

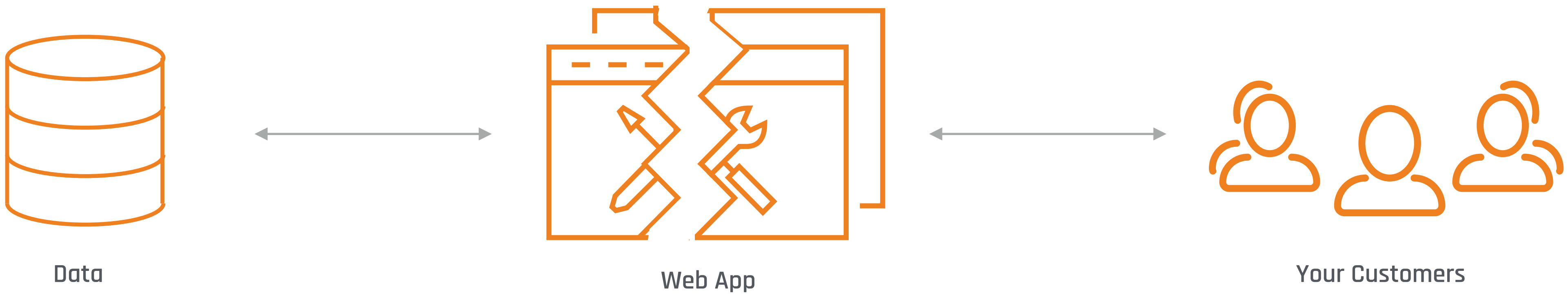
Lack of a solid architecture

ISSUES

Improvised architecture
Too basic standards
Limited use of packages
No time to build/test common features
Business logic mixed with app

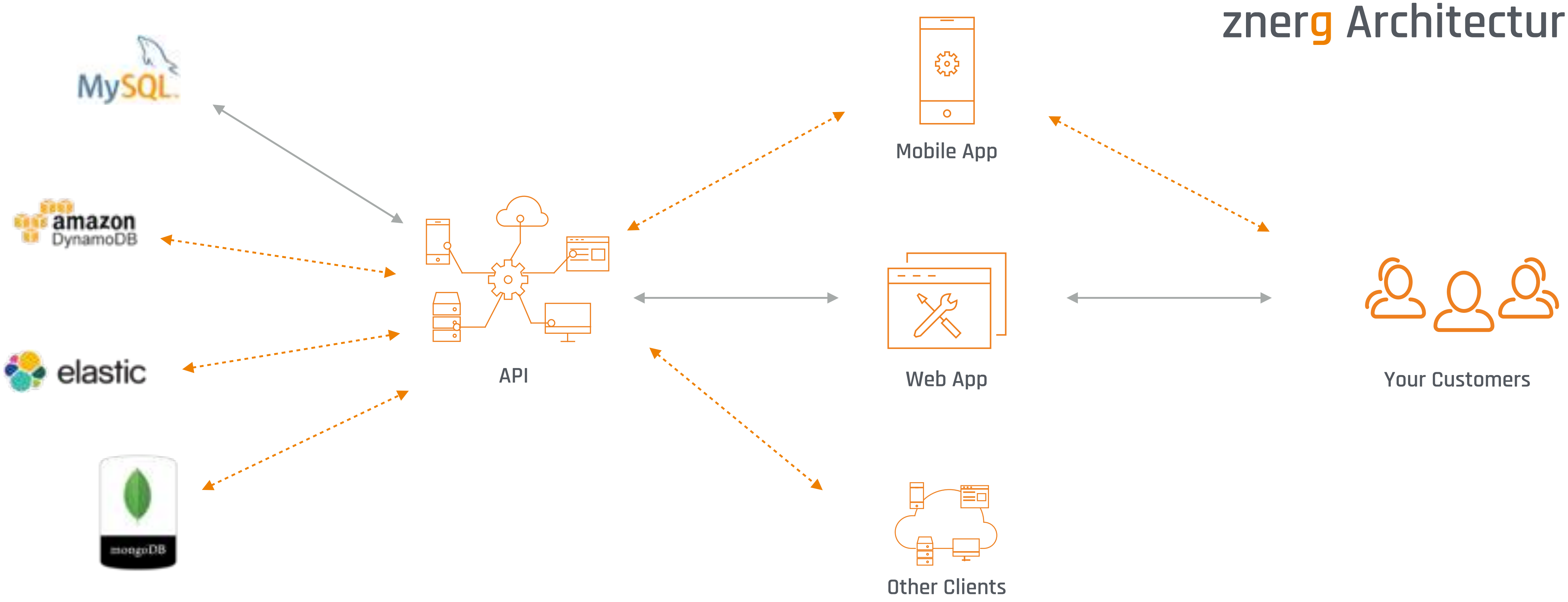
REQUIREMENTS

- ✗ **OK** User Experience
- ✗ **Low** Quality
- ✗ **Slow** Development
- ✗ **High-Cost** Maintenance
- ✗ **Hard-to-Extend**
- ✗ **Poor** Features
- ✗ **High** Price



OUR APPROACH

znerg Architecture



RESULT

Proven architecture
Standards
Many packages to reuse
Already embedded functionality
Business Logic separated in API

REQUIREMENTS

- ✓ Excellent User Experience
- ✓ High Quality
- ✓ Rapid Development
- ✓ Low-Cost Maintenance
- ✓ Easy-to-Extend
- ✓ Rich Features
- ✓ Reasonable Price

Added value:
Mobile-ready
Open to external systems
Lower Operational Cost
More Opportunities

well, all that sounds good but

IT SURE IS
COSTLY AND DIFFICULT
TO IMPLEMENT, RIGHT?

NOT AT ALL!

Q: Does this architecture require more investment?

A: No, your actual ROI will be much higher because you will have much more functionality and low-cost maintenance.

Q: Would I need to hire new, more skilful people?

A: No, not at all. We will provide guides that will help any experienced PHP developer to get productive in a few days.

Q: Is this going to take longer to develop?

A: Actually, quite the opposite.

Q: Can I modify or extend the products?

A: Yes, you are free to modify or extend any products, but at your own risk. We could execute custom modifications for you at a maintenance rate.

Q: Do I get any support for this?

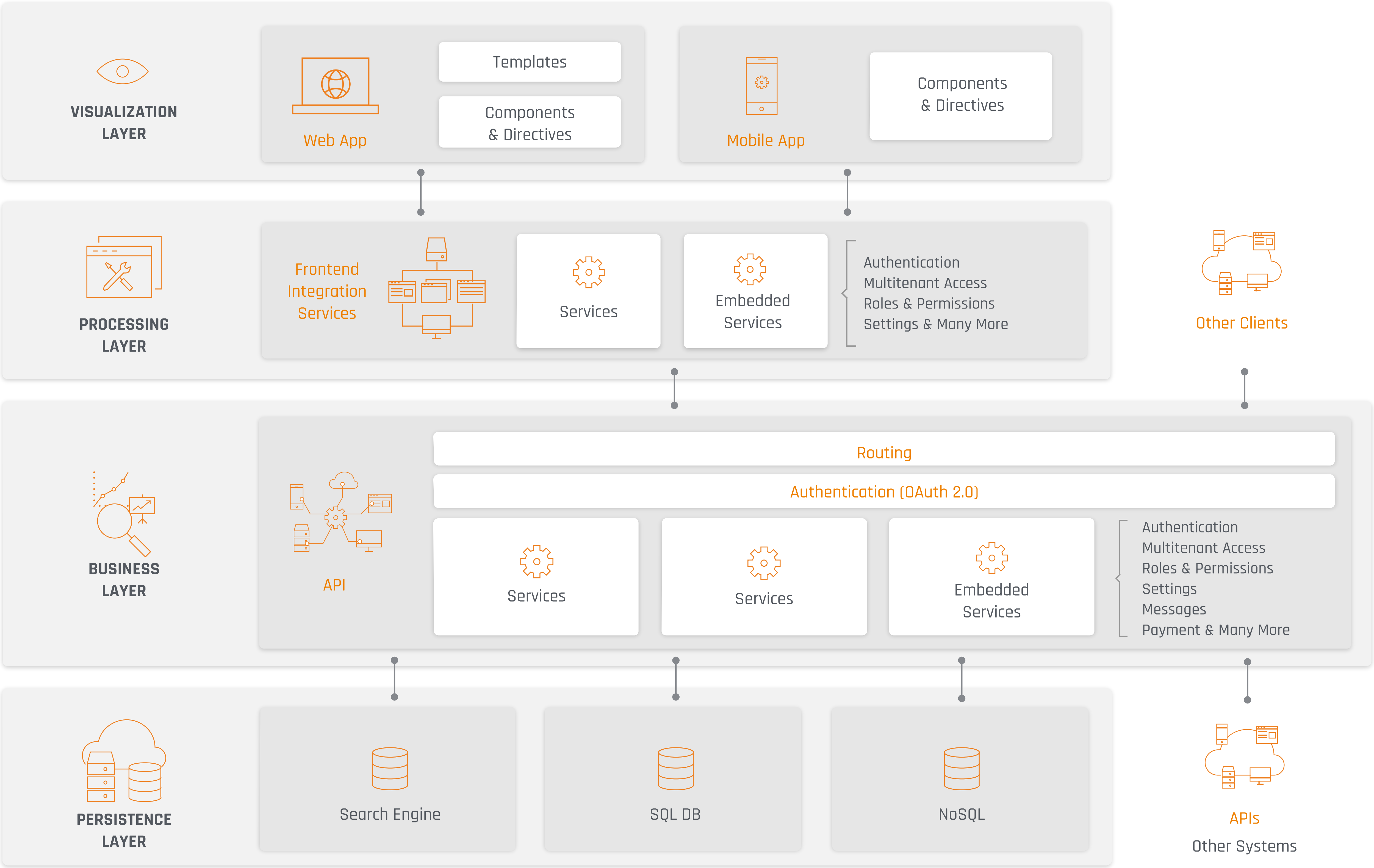
A: Yes, of course. We provide 1-year* free customer support service for any bug or issue in any of our products.

*From the day of installation in a live environment or the last day we installed the product in a customer server, whichever is more recent.

how

DO WE DO IT?

OUR ARCHITECTURE



VISUALIZATION LAYER

Web App

Templates

Components & Directives

Mobile App

Components & Directives

PROCESSING LAYER

Frontend Integration Services

Services

Embedded Services

Authentication
Multitenant Access
Roles & Permissions
Settings & Many More

Other Clients

BUSINESS LAYER

API

Routing

Authentication (OAuth 2.0)

Services

Services

Embedded Services

Authentication
Multitenant Access
Roles & Permissions
Settings
Messages
Payment & Many More

PERSISTENCE LAYER

Search Engine

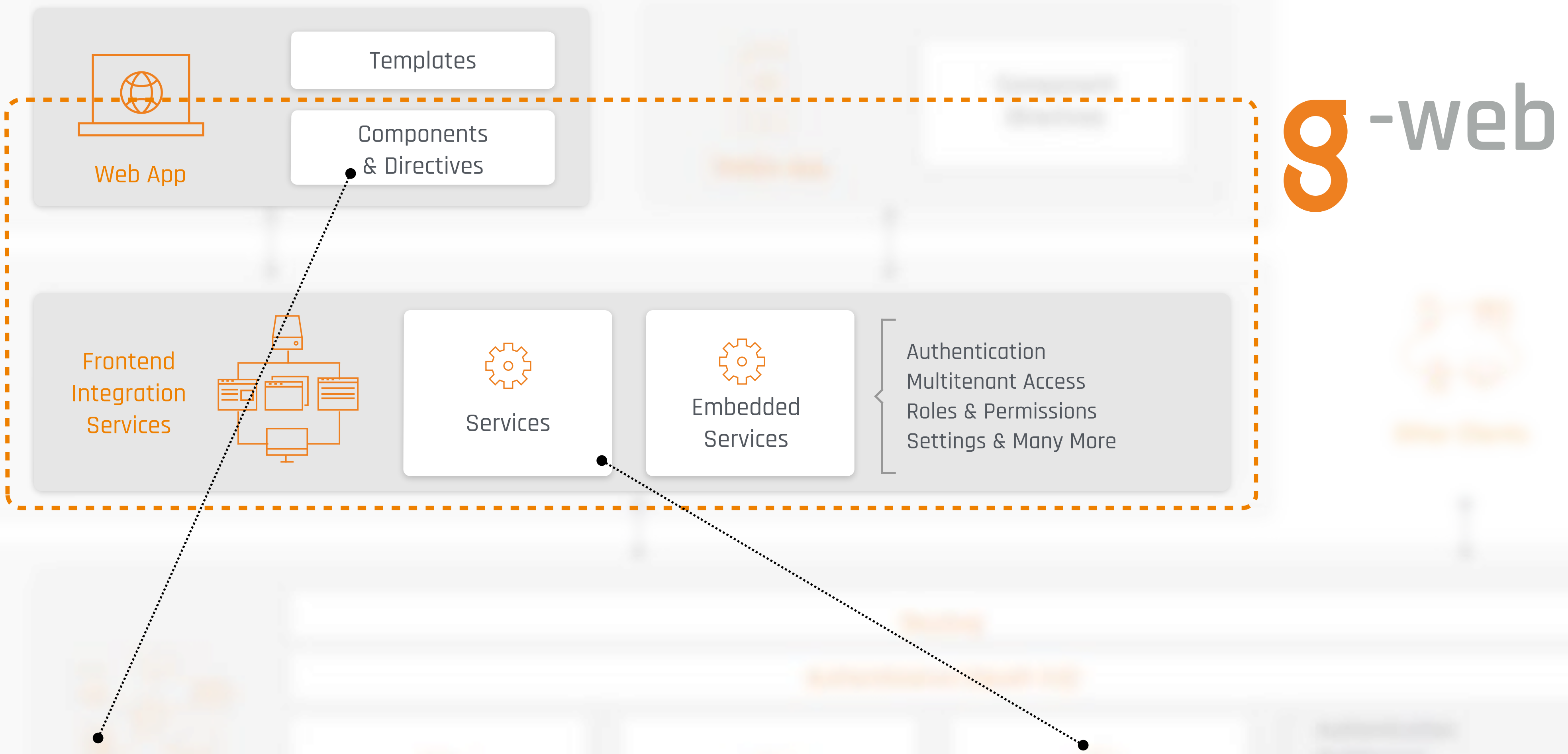
SQL DB

NoSQL

APIs

Other Systems

g-web: FRONTEND FOR DESKTOP APPS



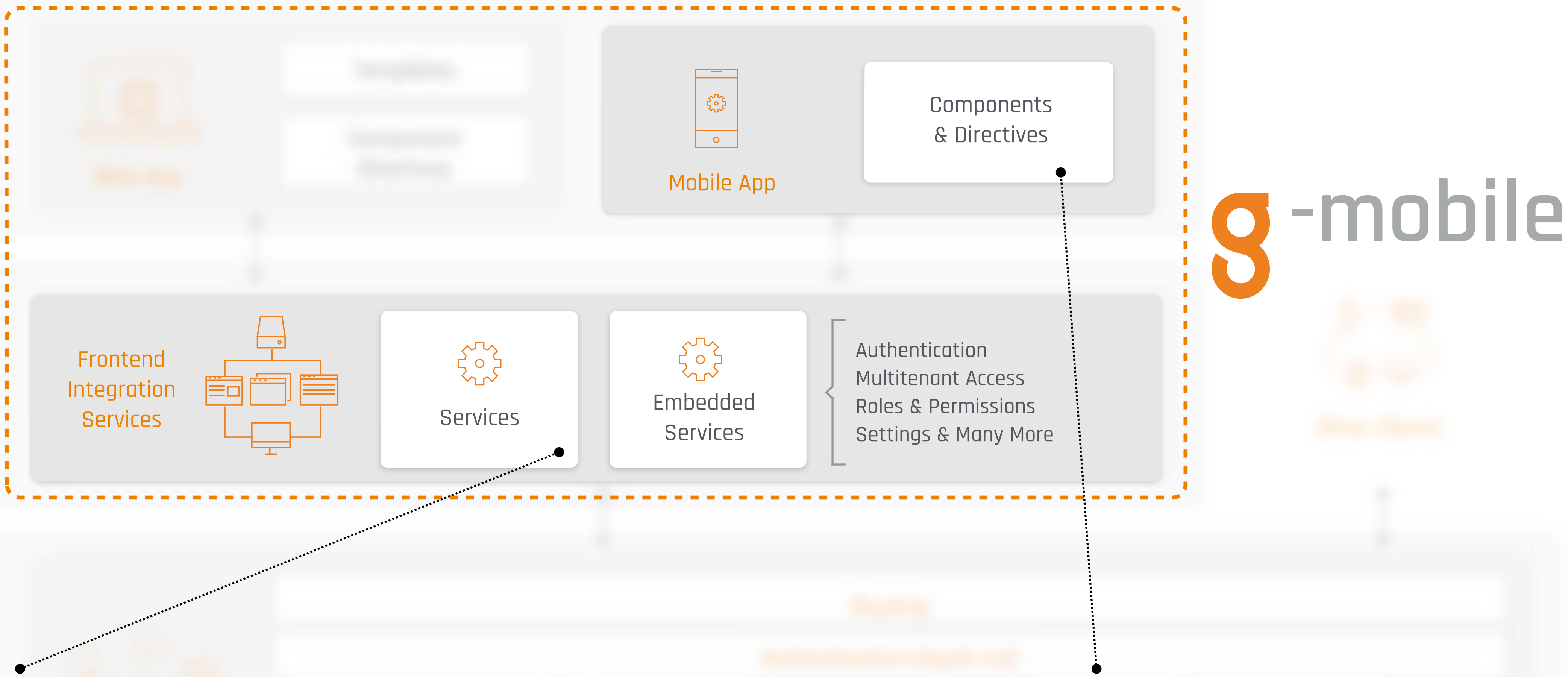
COMPONENTS/DIRECTIVES

- Sorting
- Filtering
- Pagination
- CRUD Operations
- Messages & Notifications
- Attachments
- Social Integration

FRONTEND INTEGRATION SERVICES

- Input Validator
- Error Handling
- Request Processing
  - Pagination
  - Sorting
  - Filtering
  - CRUD
  - Social Integration
- Sub-services
- Notifications
- API Processing

g-mobile: FRONTEND FOR MOBILE APPS



FRONTEND INTEGRATION SERVICES

- Input Validator
- Error Handling
- Request Processing
- Sub-services
  - Pagination
  - Sorting
  - Filtering
  - CRUD
  - Social Integration
- Notifications
- API Processing

COMPONENTS/DIRECTIVES

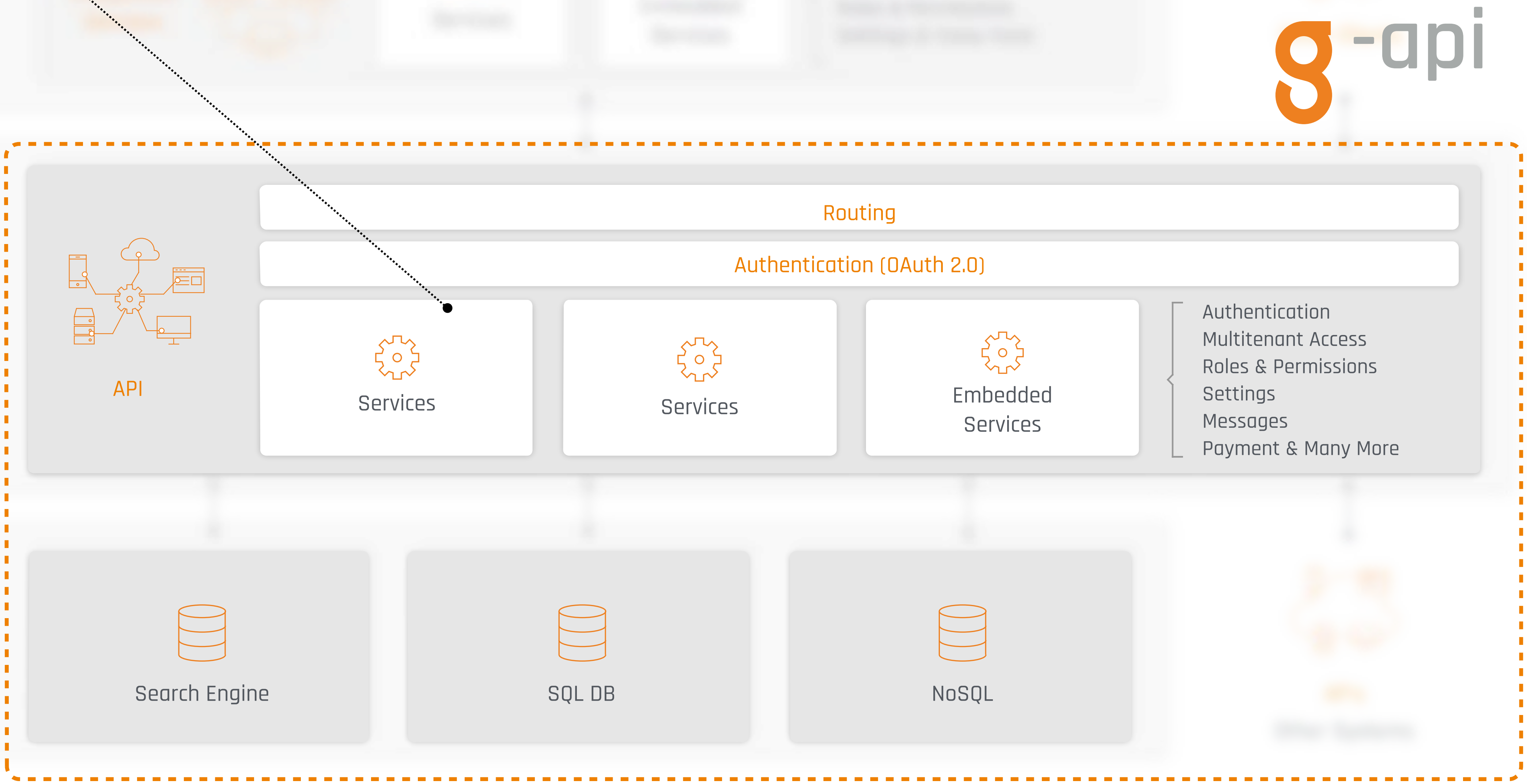
- Sorting
- Filtering
- Pagination
- CRUD Operations
- Messages & Notifications
- Attachments
- Social Integration

g-api: BACKEND FOR ENTERPRISE APPLICATIONS

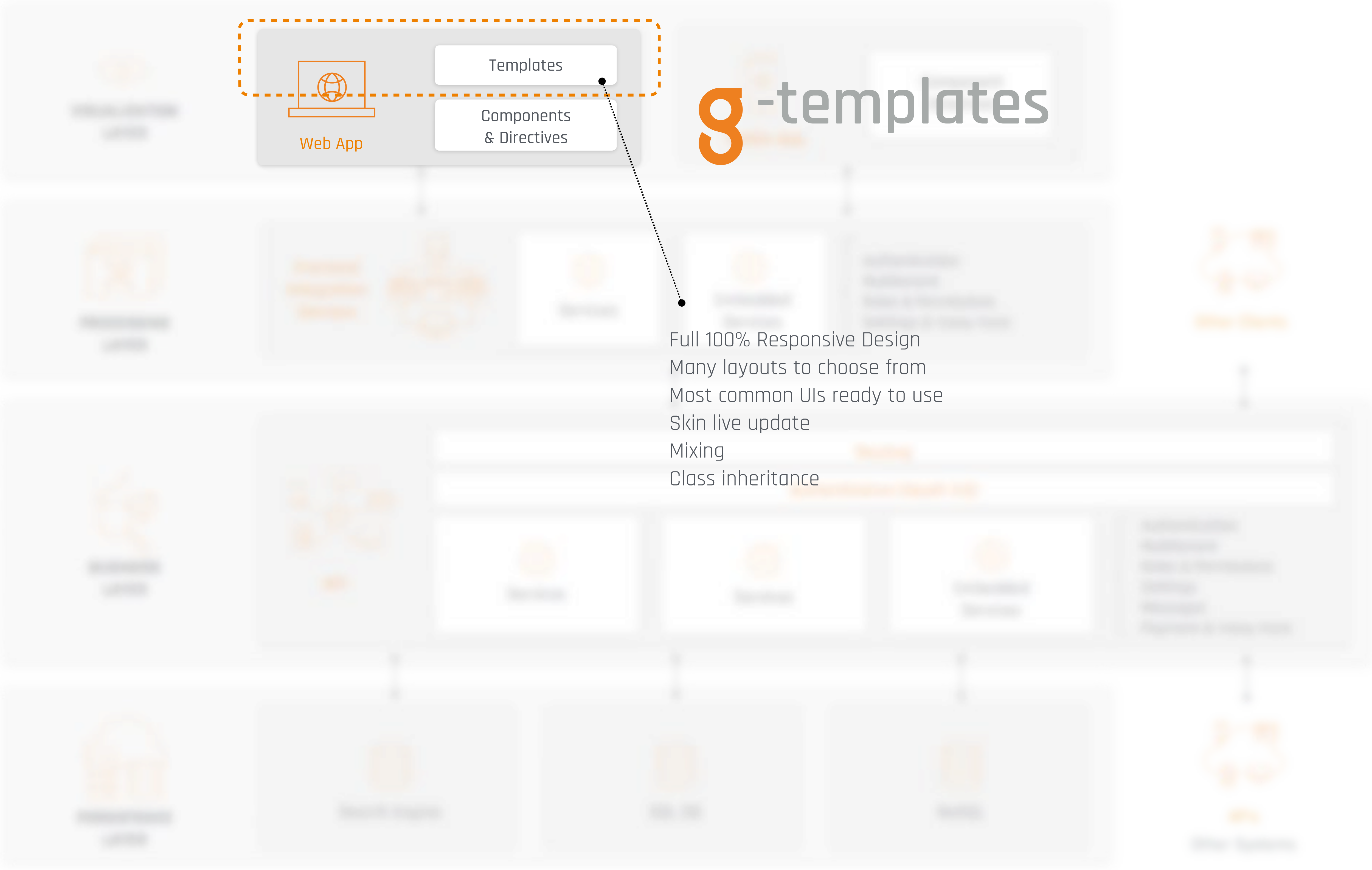
API SERVICE

Input Validator
Error Handling
Logic Processing
Sub-services
Events/Listeners/Webhooks
Repositories
JSON Transformer

Pagination
Sorting
Filtering
Transactions
CRUD



g-templates: FRONTEND FOR DESKTOP APPS



Full 100% Responsive Design
Many layouts to choose from
Most common UIs ready to use
Skin live update
Mixing
Class inheritance

but

WHAT ARE THE ACTUAL BENEFITS?

FACTS

Less coding

Fewer errors

Easier to maintain

Easier to test

Lower maintenance cost

Higher quality

Rapid development

Scalable & Extensible

Ready for mobile & other systems

More secure (OAuth 2.0)

Better performance

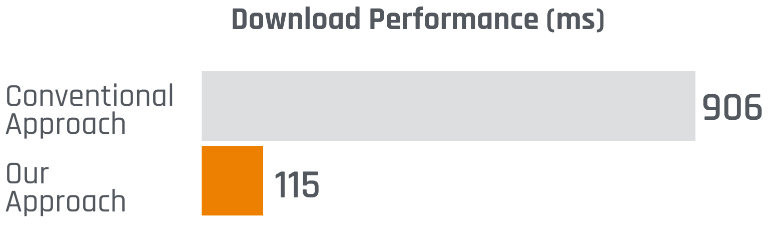
Improved user experience

More efficient (less network traffic)

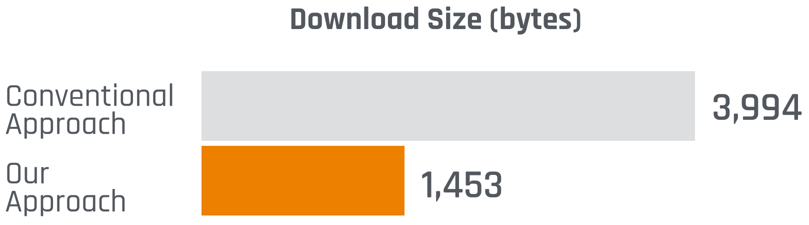
Cooler!

FIGURES

LOADING TASKS (CACHE)

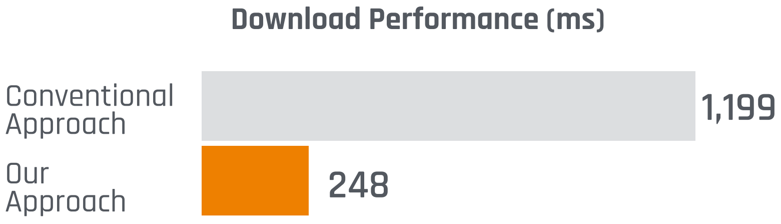


IMPROVEMENT: **87%**

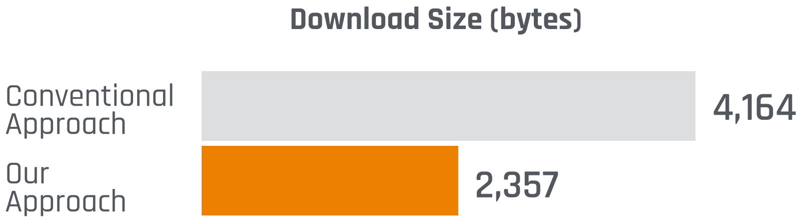


IMPROVEMENT: **64%**

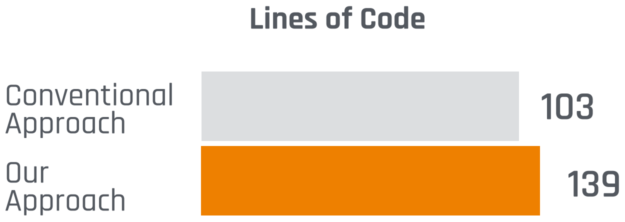
CREATING TASKS (CACHE)

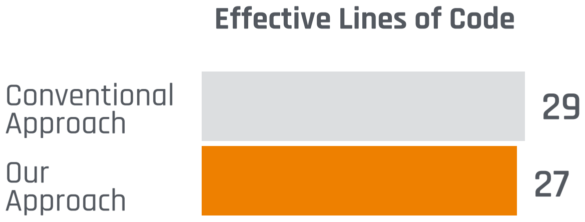


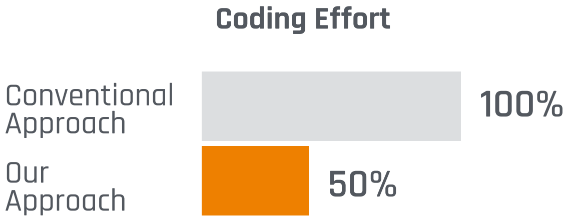
IMPROVEMENT: **79%**



IMPROVEMENT: **43%**







let's start

YOUR NEXT PROJECT TODAY

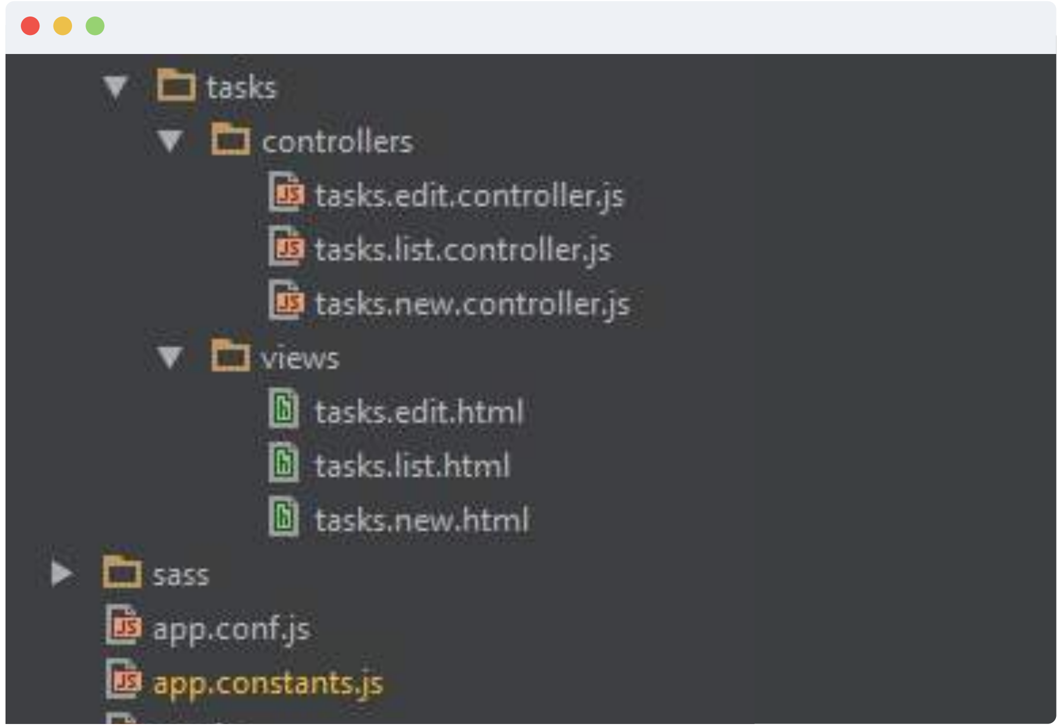
using

znerg Architecture

geek enough?

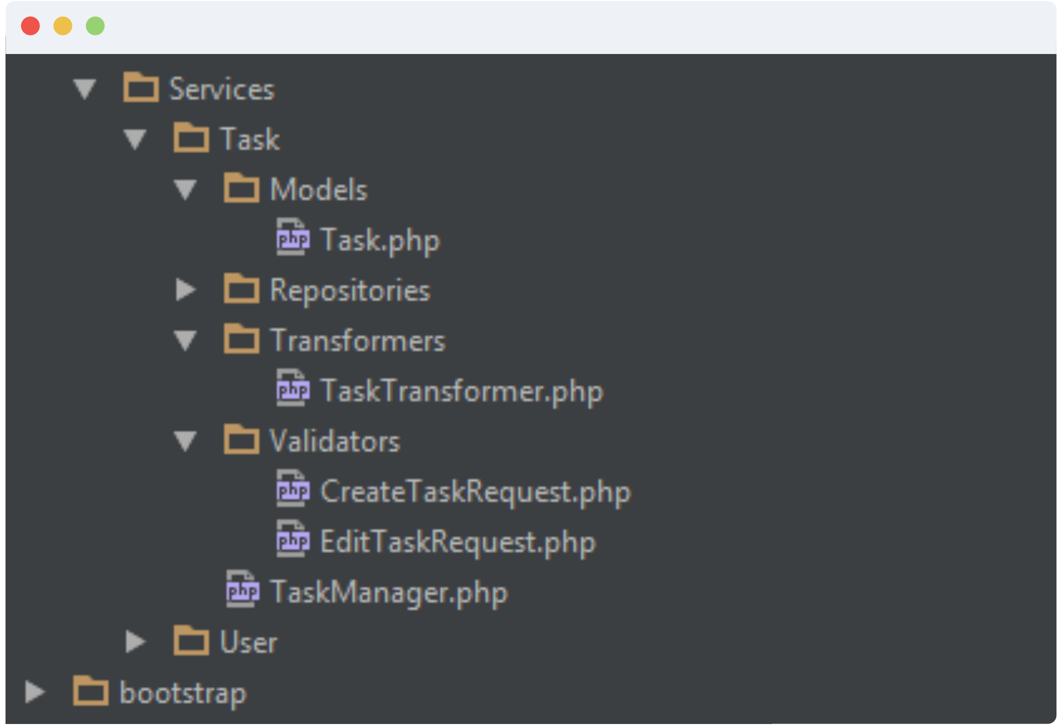
TAKE A PEEK AT OUR ARCHITECTURE'S GUTS

ARCHITECTURE



```
▼ tasks
  ▼ controllers
      tasks.edit.controller.js
      tasks.list.controller.js
      tasks.new.controller.js
  ▼ views
      tasks.edit.html
      tasks.list.html
      tasks.new.html
▶ sass
  app.conf.js
  app.constants.js
```

Frontend



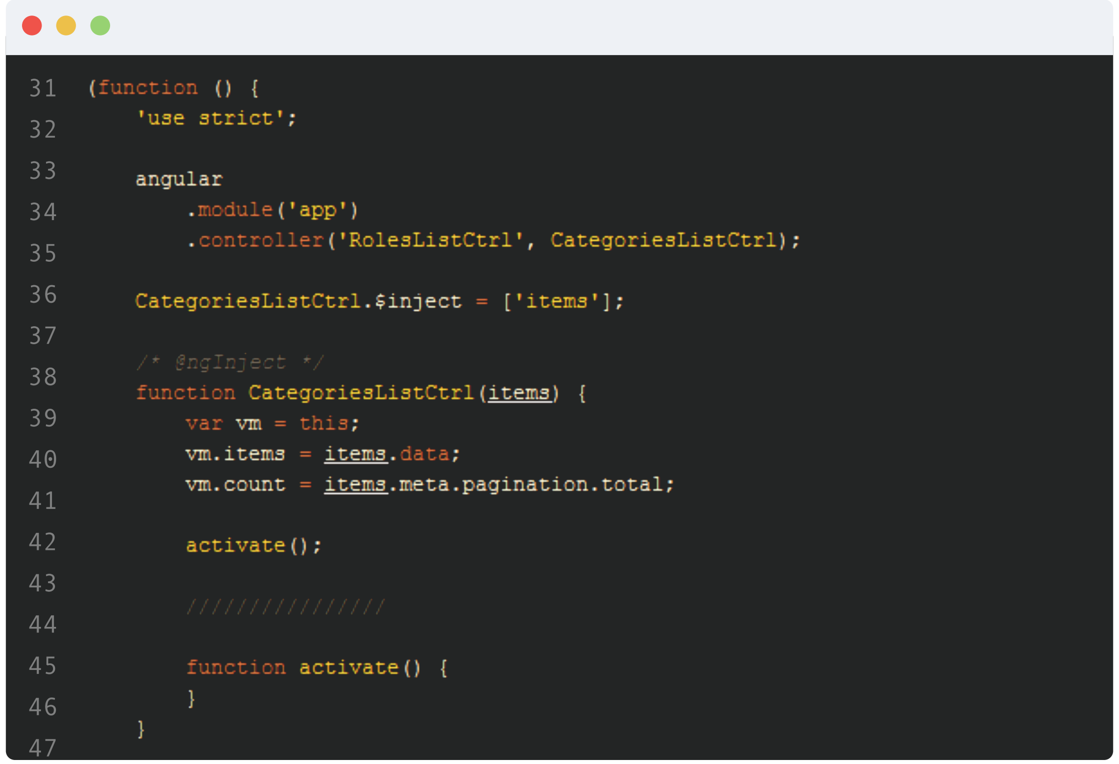
```
▼ Services
  ▼ Task
    ▼ Models
        Task.php
    ▶ Repositories
    ▼ Transformers
        TaskTransformer.php
    ▼ Validators
        CreateTaskRequest.php
        EditTaskRequest.php
      TaskManager.php
  ▶ User
▶ bootstrap
```

Backend

CODING A CRUD

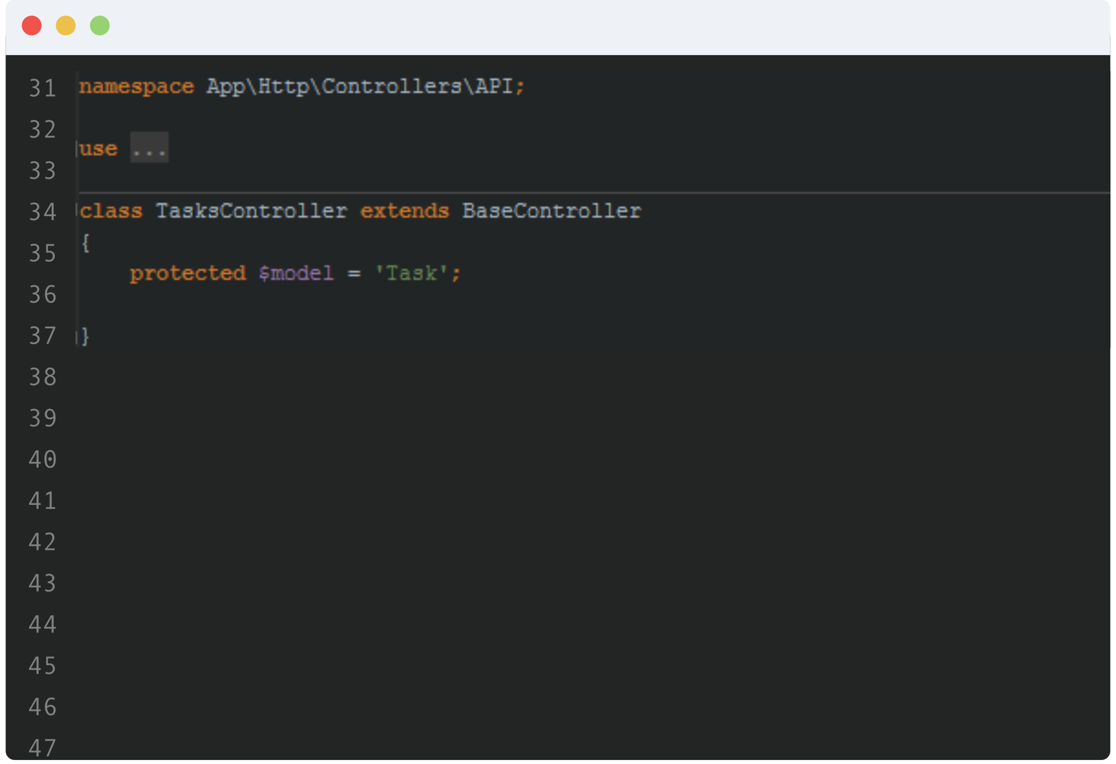
Using znerg Architecture

No Coding



```
(function () {
    'use strict';

    angular
        .module('app')
        .controller('RolesListCtrl', CategoriesListCtrl);

    CategoriesListCtrl.$inject = ['items'];

    /* @ngInject */
    function CategoriesListCtrl(items) {
        var vm = this;
        vm.items = items.data;
        vm.count = items.meta.pagination.total;

        activate();

        ////////////////

        function activate() {
        }
    }
```

Frontend



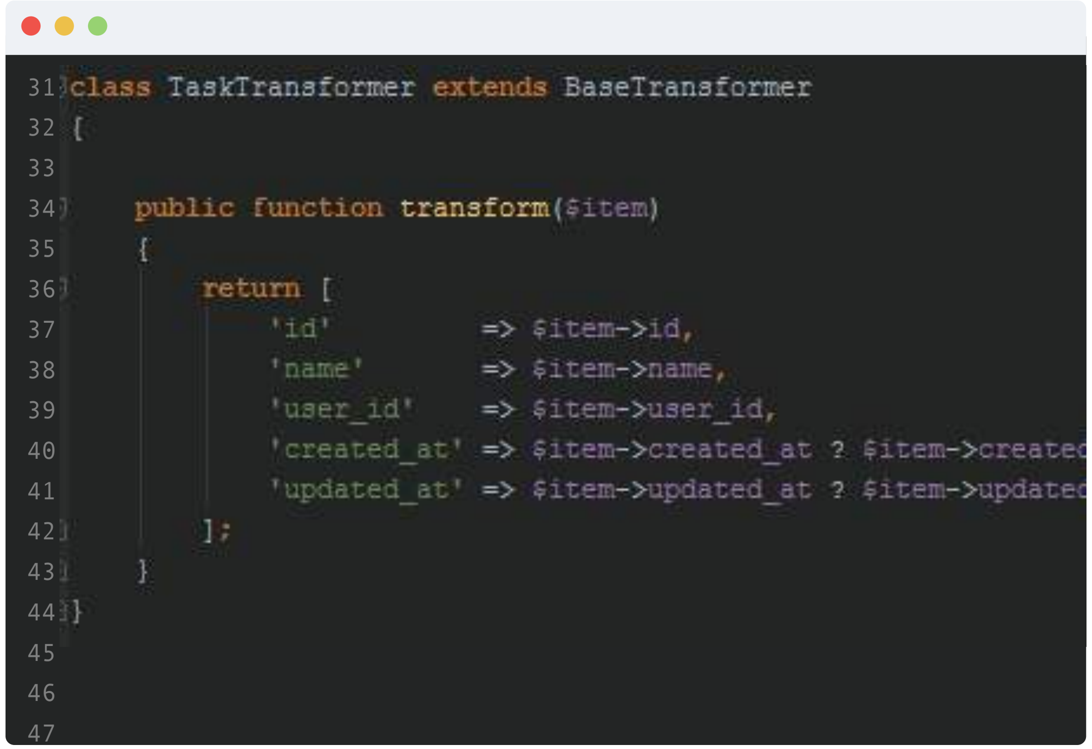
```
namespace App\Http\Controllers\API;

use ...

class TasksController extends BaseController
{
    protected $model = 'Task';

}
```

Backend

CODING A CRUD

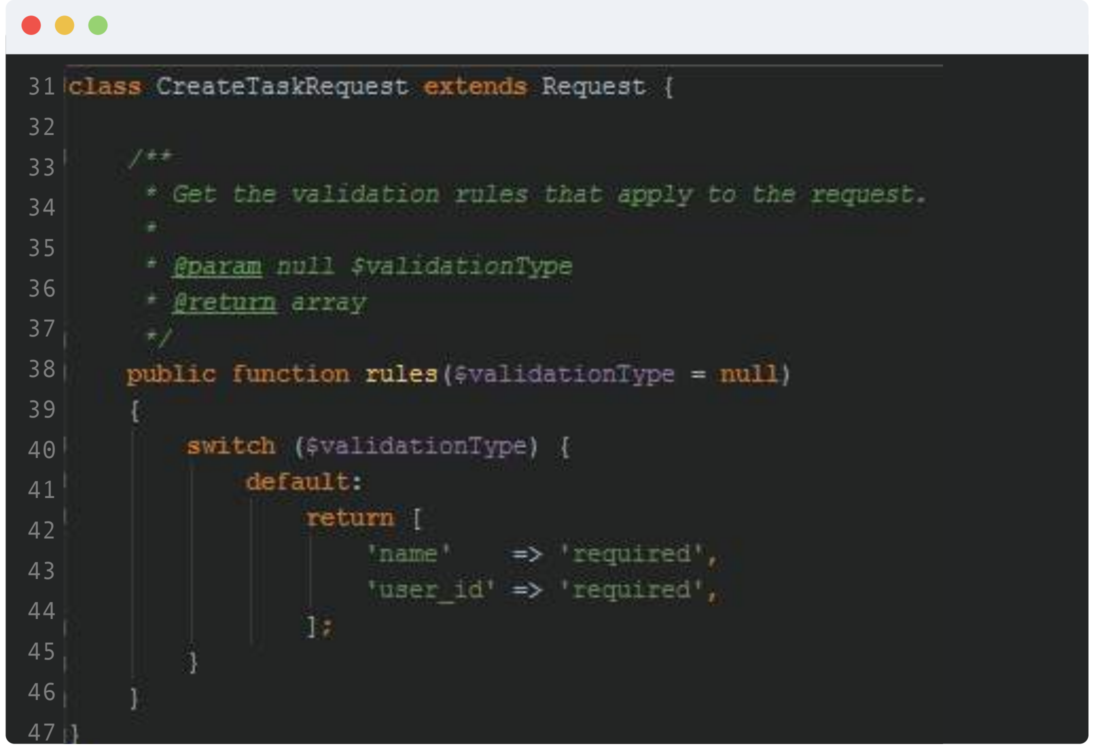
Using znerg Architecture

Configuration Only



```
class TaskTransformer extends BaseTransformer
{

    public function transform($item)
    {
        return [
            'id'         => $item->id,
            'name'       => $item->name,
            'user_id'    => $item->user_id,
            'created_at' => $item->created_at ? $item->createc
            'updated_at' => $item->updated_at ? $item->updatec
        ];
    }
}
```

Frontend



```
class CreateTaskRequest extends Request {

    /**
     * Get the validation rules that apply to the request.
     *
     * @param null $validationType
     * @return array
     */
    public function rules($validationType = null)
    {
        switch ($validationType) {
            default:
                return [
                    'name'    => 'required',
                    'user_id' => 'required',
                ];
        }
    }
}
```

Backend

This can be generated in <2 min

where
VISION and PASSION
meet
TECHNOLOGY