

METEOR KITCHEN COOKING!

@BESUTOME



Meteor Kitchen

Code generator for Meteor.js
Less coding = more code

Get started »

METEOR KITCHEN

What is Meteor Kitchen? Meteor Kitchen is a code generator for Meteor.js applications. It generates a complete Meteor application with a database (MongoDB), a REST API (JSON), a user interface (UI) and a Meteor-kitchen boilerplate. It is a great tool for prototyping and for building a complete application quickly.

- Layout
- Pages (with content)
- Menus
- Forms (with validations)
- DataViews (components you can use to show data from collections with search & sort functions)

And more: in addition to built-in components you can write your own **plugins** in javascript and expand the generator to infinity!

Did you know?

This site is a meteor application generated automatically by Meteor Kitchen. See source code (input file for generator) [here](#).

Save time & money

Build code - in minutes instead of hours. Or in hours instead of days.
In some cases you can build a complete application without writing a single line of code!

Perfect for prototyping

1. Talk to your customer
2. Draw a sitemap
3. **Write an application description file**
4. Run the generator and you have a working prototype in minutes!

High quality code

Write once - use many times

WHAT'S THE METEOR KITCHEN?

```
$ curl http://www.meteor-kitchen.com/install | /bin/sh  
$ vim myapp.json  
$ meteor-kitchen myapp.json myapp  
$ cd myapp && meteor
```

COMPLETE!

SAVE TIME & MONEY

PERFECT FOR PROTOTYPING

HIGH QUALITY CODE

WRITE ONCE - USE MANY TIMES

```
{
  "application": {
    "free_zone": {
      "pages": [
        { "name": "home", "title": "Home page" },
        { "name": "about", "title": "About" }
      ],

      "components": [
        {
          "name": "main_menu",
          "type": "menu",
          "items": [
            { "title": "Home page", "route": "home" },
            { "title": "About", "route": "about" }
          ]
        }
      ]
    }
  }
}
```

INPUT FILE STRUCTURE

- » `application.pages.name (home)`
 - » `home.html, home.js, home_controller.js`
- » filenames and route names are snake_case
- » template names are CamelCase

FRONTEND FRAMEWORK

```
{  
  "application": {  
    "frontend": "bootstrap3"  
  }  
}
```

```
{  
  "application": {  
    "frontend": "semantic-ui"  
  }  
}
```

```
{  
  "application": {  
    "frontend": "materialize"  
  }  
}
```

COLLECTIONS

```
{
  "collections": [
    {
      "name": "notes",
      "before_insert_code": "doc.name = Meteor.user() ? Meteor.user().profile.name : 'Anonymous'; return doc;",
      "owner_field": "createdBy",
      "roles_allowed_to_read": [],
      "roles_allowed_to_insert": [],
      "roles_allowed_to_update": [
        "owner"
      ],
      "roles_allowed_to_delete": [
        "owner"
      ]
    }
  ]
}
```


QUERIES

```
{
  "queries": [
    {
      "name": "current_user_data",
      "collection": "users",
      "filter": {
        "_id": "Meteor.userId()"
      },
      "find_one": true,
      "options": {}
    }
  ]
}
```

PAGES

```
{ "pages": [ {
  "name": "home_free",
  "text": "",
  "components": [
    {
      "name": "photo_button",
      "html": "",
      "type": "custom_component"
    },
    {
      "name": "form",
      "mode": "insert",
      "submit_route": "home_free",
      "type": "form",
      "query_name": "notes_empty",
      "fields": [
        {
          "name": "text",
          "title": "Or enter a note:",
          "type": "string",
          "required": true,
          "input": "textarea"
        }
      ]
    }
  ]
} ] }
```

HTML

title

```
<h5>Hey developer, programming all day (and night) long?</h5>
<h4>Show us your tired face! :)</h4>
```

html

```
<template name=\"TEMPLATE_NAME\">\n  {{#each notes}}\n    \t<div class=\"panel\" style=\"padding: 0 10px;\">\n      {{#if photo}}\n        <img src=\"{{photo}}\" alt=\"Loading image...\" style=\"margin-top: 10px; width: 100%; height: auto\">\n      {{else}}\n        <p>{{text}}</p>\n      {{/if}}\n      <p class=\"text-muted\">{{name}}, {{livestamp createdAt}}</p>\n    </div>\n  {{/each}}\n</template>
```

COMPONENTS

```
{
  "components": [
    {
      "name": "side_menu",
      "type": "menu",
      "class": "nav nav-stacked nav-pills",
      "items": [
        {
          "title": "Profile",
          "route": "user_settings.profile"
        },
        {
          "title": "Change password",
          "route": "user_settings.change_pass"
        }
      ]
    }
  ]
}
```

SUMMARY

» json only

» auto generate

Let's Cooking

LEURKING

