

```
meteor npm install --save vue
meteor npm install --save-exact @babel/runtime@7.0.0-beta.55
meteor add akryum:vue-component
```

```
npm i quasar-framework
```

```
npm i quasar-extras
```

```
ln -s ../node_modules/quasar-framework imports
```

```
npm install vue-router --save
```

```
npm install vee-validate --save
```

```
validated method
```

```
meteor add mdg:validated-method
```

```
meteor add didericis:callpromise-mixin
```

```
meteor add aldeed:simple-schema
```

```
meteor add aldeed:collection2@3.0.0
```

```
meteor npm install --save simpl-schema
```

```
npm install simpl-schema --save
```

```
create collection
```

```
https://github.com/aldeed/meteor-collection2#installation
```

```
import {Mongo} from 'meteor/mongo';

import SimpleSchema from 'simpl-schema';
export default Item = new Mongo.Collection('items')
const Item = new Mongo.Collection('items')
let schema={ }
schema.items=new SimpleSchema({
  name:{
    type:String,
    min:3,
    max:20,
  },
  price:{
    type:Number,
    min:1,
  },
},
```

```

cust: {
  type: Number,
  min: 0,
},
memo: {
  type: String,
  optional: true,
}
})

```

```
Item.attachSchema(schema.items);  
export default Item;
```

## Create validate method

```
import {Meteor} from 'meteor/meteor';
```

```
import Item from '../collections/item';import SimpleSchema from "simpl-schema";
```

```
export const insertItem = new ValidatedMethod({
  name: 'insertItem',
  mixins: [CallPromiseMixin],
  validate: new SimpleSchema({
    name: {
      type: String,
      max: 100
    },
    price: {
      type: Number,
      min: 1
    },
    cust: {
      type: Number,
      min: 1
    },
    memo: {
      type: String,
      max: 200,
      optional: true
    }
  }), ស្គាល់ហើយរសជាតិដកធ្មេញ ថាអត់ឈឺមា
```

```
}).validator(),
run(doc) {
  return Item.insert(doc)
}
});

export const findItem = new ValidatedMethod({
  name: 'mfindItem',
  mixins: [CallPromiseMixin],
  validate: null,
```

```

run() {
  return Item.find().fetch();
}
});
export const removeItem = new ValidatedMethod({
  name: 'removeItem',
  mixins: [CallPromiseMixin],
  validate: new SimpleSchema({
    _id: String,
  }).validator(),
  run(_id) {
    return Item.remove(_id)
  }
});
export const updateItem = new ValidatedMethod({
  name: 'updateItem',
  mixins: [CallPromiseMixin],
  validate: new SimpleSchema({
    _id: {
      type: String
    },
    name: {
      type: String,
      max: 100
    },
    price: {
      type: Number,
      min: 1
    },
    cust: {
      type: Number,
      min: 1
    },
    memo: {
      type: String,
      max: 200,
      optional: true
    },
  }).validator(),
  run(doc) {
    return Item.update({
      _id: doc._id
    }, {
      $set: doc
    })
  }
});
export const findItemById = new ValidatedMethod({
  name: 'mfindItemById',
  mixins: [CallPromiseMixin],

```

```

validate: new SimpleSchema({
  id: String
}).validator(),
run(_id) {
  return Item.findOne(_id)
}
})

```

npm install moment --save

npm install numeral

ដើម្បីប្រើប្រាស់ Aggregate បានល្អៗត្រូវតែដំឡើង sakulstra:aggregate

meteor add sakulstra:aggregate

## Usage

```

let data = Invoice.aggregate(
  [{
    $lookup: {
      from: 'customers',
      localField: 'customerId',
      foreignField: '_id',
      as: 'customerDoc',
    }
  }]
)
return data

```

## ចង់Loop ពេលធ្វើReport ទម្រង់

```

<tbody>

<template v-for="(doc,i) in reports">
<template v-for="(item,itemIndex) in doc.items">
<tr
v-if="itemIndex==0"
:key="item.itemId">
<td>{{doc.date | getDate}}</td>
<td>
<router-link :to="{name:'invoiceForm',params:{_id:doc._id,
name:'reportByCustomer'}}">{{i}}</router-link>
</td>
<td>{{doc.customerName}}</td>

```

```

<td>{{ item.itemName }}</td>
<td>{{ item.qty }}</td>
<td>{{ item.price | getMoney }}</td>
<td>{{ item.amount | getMoney }}</td>
</tr>
<tr
v-else
:key="item.itemId">
<td colspan="3"></td>
<td>{{ item.itemName }}</td>
<td>{{ item.qty }}</td>
<td>{{ item.price | getMoney }}</td>
<td>{{ item.amount | getMoney }}</td>
</tr>
</template>
<tr :key="doc._id">
<td colspan="6"></td>
<td>{{ doc.total | getMoney }}</td>
</tr>
</template>

</tbody>

```

ចង់ប្រើ Button Back

```

<q-btn
class="float-right"
color="info"
icon="undo"
@click="$router.back()"
/>

```

npm=>vue-meteor-tracker វាចេះ React ដោយខ្លួនឯងមិនបាច់ប្រើ Methods មក Refresh

meteor npm install --save vue-meteor-tracker

```
import VueMeteorTracker from 'vue-meteor-tracker'
```

```
Vue.use(VueMeteorTracker)
```

ត្រូវទៅមើល Aggregate បន្ថែមត្រង់ចំណុច Report by date