2019-03-31工作记录-创建众筹实例

当点击“查看众筹”时跳转到众筹的详情页面，需要知道这个众筹活动的详细数据（管理者、最小捐款金额，当前账户金额，捐款人数量，提款请求数量等等），这些数据从何而来？需要从区块链上已经发布的众筹合约地址读取数据，最好是把该众筹合约地址变成一个众筹对象实例，方便我们读写。本文记录如何创建一个众筹实例，设计html页面显示出来。

第一部分，创建一个实例，得到详细数据

1 创建文件ethereum/campaign.js, 能根据一个合约地址创建一个实例。

//创建众筹活动的实例模块， 只要知道这个众筹合约的地址就能创建出来一个实例。

import web3 from './web3';

import Campaign from './build/Campaign.json';

export default (address)=>{

const instance = new web3.eth.Contract(

JSON.parse(Campaign.interface),

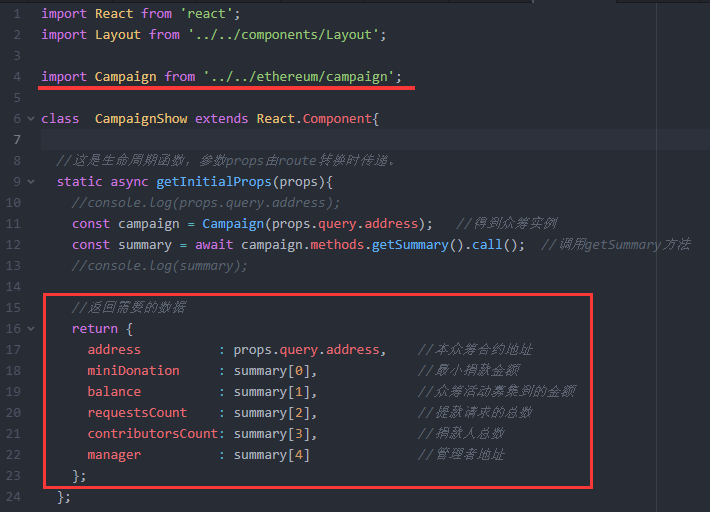
address

);

return instance;

}

2 修改pages/campaigns/show.js, 增加详情数据读取代码：

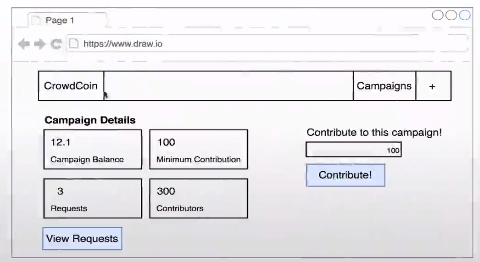


注意在render函数中： 从props参数获取这些数据。

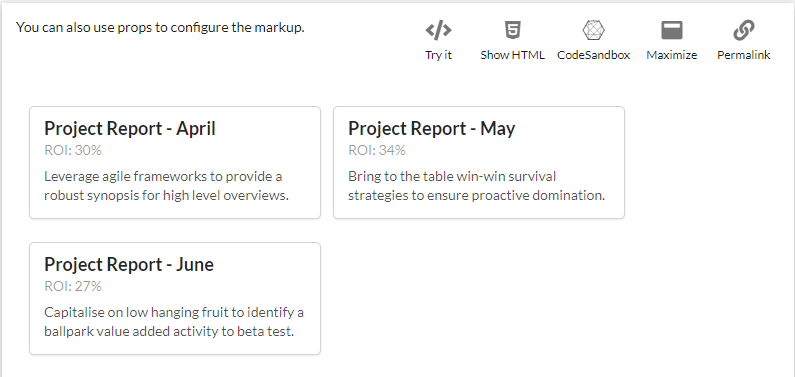


第二部分，设计页面，显示出数据。

1 界面示意：



2 选择使用react-UI的card控件样式，效果如图：



修改show.js, 导入Card，定义函数

import React from 'react';

import Layout from '../../components/Layout';

import Campaign from '../../ethereum/campaign';

import { Card } from 'semantic-ui-react';

class CampaignShow extends React.Component{

//这是生命周期函数，参数props由route转换时传递。

static async getInitialProps(props){

//console.log(props.query.address);

const campaign = Campaign(props.query.address); //得到众筹实例

const summary = await campaign.methods.getSummary().call(); //调用getSummary方法

//console.log(summary);

//返回需要的数据

return {

address : props.query.address, //本众筹合约地址

miniDonation : summary[0], //最小捐款金额

balance : summary[1], //众筹活动募集到的金额

requestsCount : summary[2], //提款请求的总数

contributorsCount: summary[3], //捐款人总数

manager : summary[4] //管理者地址

};

}

//定义控件函数

renderCards(){

//属性赋值语句，快速写法

const {

address,

miniDonation,

balance,

requestsCount,

contributorsCount,

manager

} = this.props;

const items = [

{

header: manager,

description: '管理者创建了当前众筹活动，并且是众筹的受益人',

meta: '管理者地址',

style:{overflowWrap:'break-word'} //长地址自动换行

}

];

//必须要返回items对象，不然无法显示出来

return <Card.Group items={items} />

}

render(){

return (

<Layout>

<h1>众筹详情</h1>

{this.renderCards()}

</Layout>

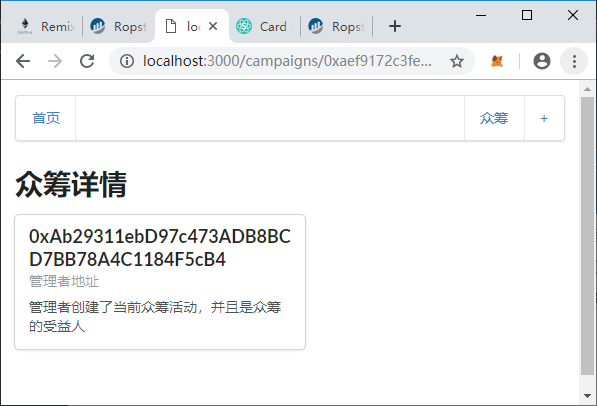
);

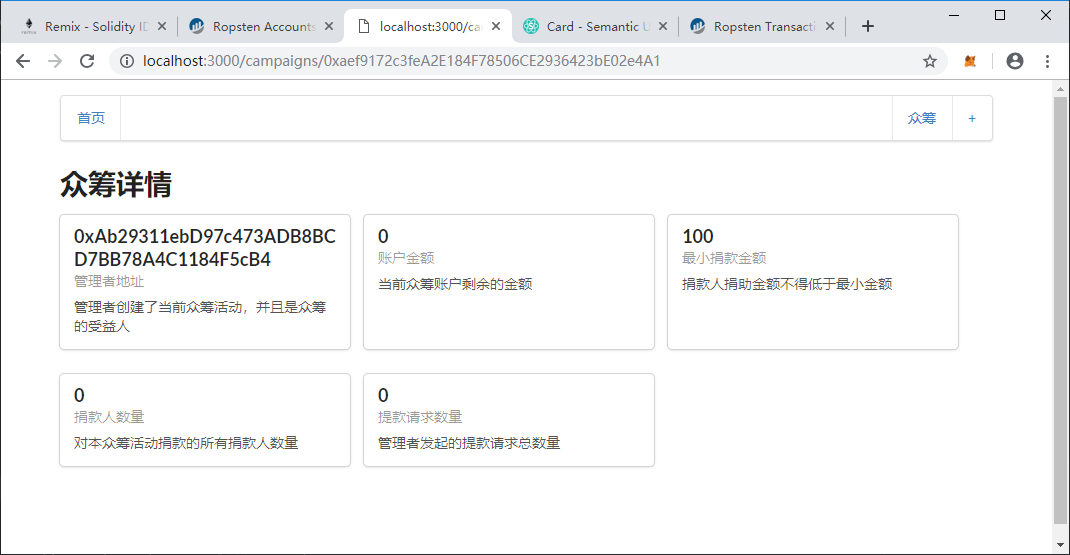
}

}

export default CampaignShow;

现在的实际效果如下，显示了管理者元素，还可以添加更多元素，显示出更多内容。





第三部分 在详情页面上添加捐款操作控件

1 用react-UI的Form表单定义出自己的捐款操作表单。

创建文件components/contributeForm.js,

import React , {Component} from 'react';

import { Form,Button,Input } from 'semantic-ui-react';

class ContributeForm extends Component{

render(){

return (

<Form>

<Form.Field>

<label>请输入您的捐款金额</label>

<Input label='wei' labelPosition='right' />

</Form.Field>

<Button primary>捐款</Button>

</Form>

);

}

}

export default ContributeForm;

2 修改show.js, 显示出来捐款操作表单

import ContributeForm from '../../components/contributeForm';

render(){

return (

<Layout>

<h1>众筹详情</h1>

{this.renderCards() }

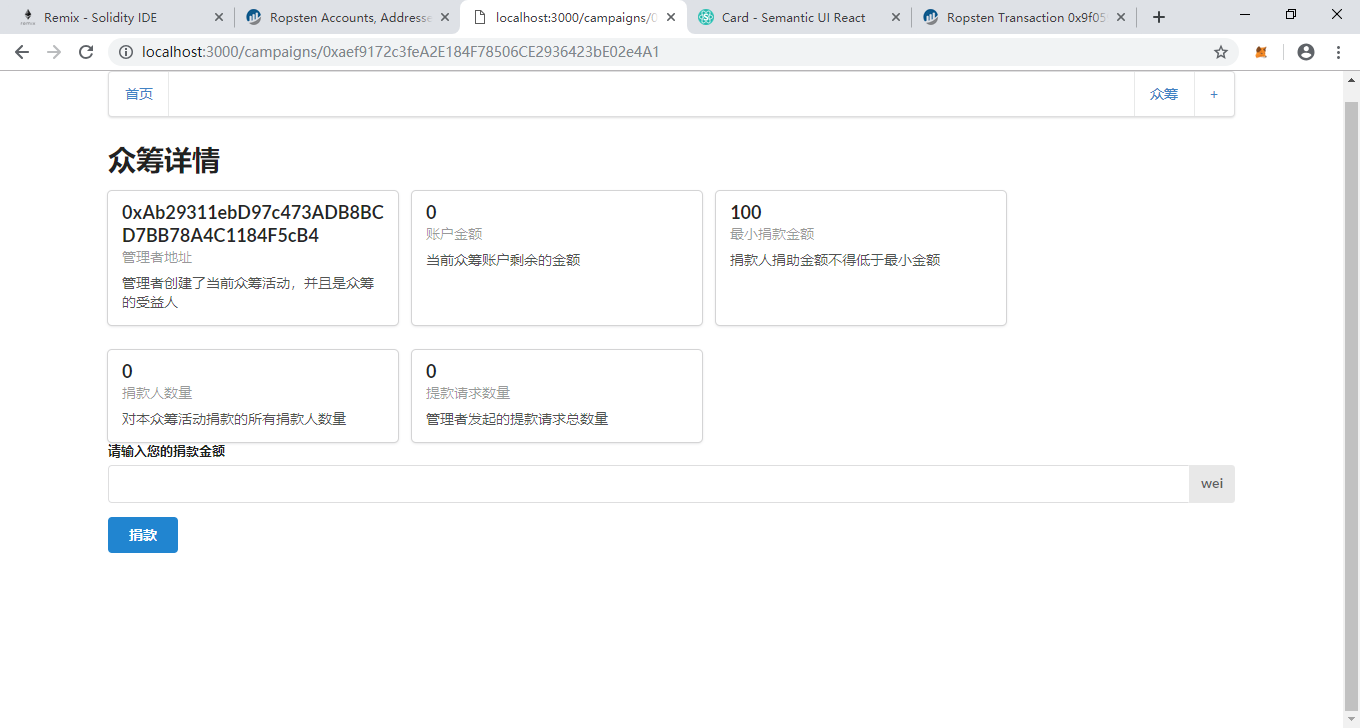
<ContributeForm />

</Layout>

);

}

效果如下图， 明显的这个捐款form放在最下方不合适，我们想放在右半边，需要用到Grid布局，接下来我们实现它。



第四部分 用Grid布局设置详情和捐款表单。

修改show.js, 导入Grid标签。

import { Grid } from 'semantic-ui-react';

修改render函数：

render(){

return (

<Layout>

<h1>众筹详情</h1>

<Grid>

<Grid.Column width={10}>

{this.renderCards() }

</Grid.Column>

<Grid.Column width={6}>

<ContributeForm />

</Grid.Column>

</Grid>

</Layout>

);

}

效果如下图

