



BACK-END DEVELOPMENT



BACKEND DEVELOPERS



EXPRESS FRAMEWORK

Фреймворк Express: Роутинг

app.[get/post/put/...] – позволяет определить процесс обработки и результат соответствующего запроса

```
var app = express();
var path = require('path');
app.get('/', function(req, res) {
  //Join all arguments together and normalize the resulting path.
  res.sendFile(path.join(__dirname + '/client', 'index.html'));
});
```

Фреймворк Express: Подключение ресурсов

app.use – Добавляет прослойку обработки запросов

express.static – Подкачивает файлы в указанной папке сервера к определенной странице на клиенте

```
app.use(express.static(__dirname + '/client'));  
app.use('/client', express.static(__dirname + '/client'));
```

Фреймворк Express: Запуск сервера

http.Server(app) – Создает сервер для соответствующей функции обработки запросов (в нашем случае – фреймворка Express)

http.listen(port) – Запускает сервер на заданном порту и начинает принимать запросы

```
var http = require('http').Server(app);  
http.listen(3000, function(){  
  console.log('Web server Active listening on *:3000');  
});
```



JS FRONTEND INTERACTIVITY

Frontend Javascript: Доступ к элементам HTML

document.getElementById('ID') – позволяет получить доступ к объекту, соответствующему элементу HTML по ID

elem.style – доступ к стилю элементу

elem.innerHTML – доступ к содержимому элемента

```
<p id="Action">Connecting...</p>
<script>
  var mytext = document.getElementById('Action');
  mytext.innerHTML = "Connecting...";
</script>
```


Frontend Javascript: Обработка событий

- **click** –клик на элемент
- **mouseover** – возникает когда на элемент наводится мышь
- **mousemove** – движение мыши
- **submit** – отправка формы `<form>`
- **focus** – фокус на элементе

```
document.getElementById('myElement').onclick = function() {  
    alert('Спасибо')  
}
```



SOCKETS.IO

Библиотека Sockets: Формат вызовов

- **Название вызова**
- **Сообщение в формате JSON**

```
("updateProfile",  
{username: "Vasya",  
password: "qwer543",  
picture: "vas1232bh.jpg"})
```

Библиотека Sockets: Обработка событий, фронтенд

Инициализация:

```
var socket = io();
```

Вызов события:

```
socket.emit('toggle led', {value: 0, userId: userId});
```

Обработка события:

```
socket.on('toggle led', function(msg) {  
    document.getElementById('text').innerHTML = msg.userId;  
});
```

Библиотека Sockets: Обработка событий, бэкенд

Инициализация:

```
var http = require('http').Server(app);  
var io = require('socket.io')(http);  
io.on('connection', function(socket) {  
    [CODE]  
});
```

Объект сокета создается один раз, поэтому стоит вести учет пользователей! Добавление в `io.on('connection')`, удаление в `window.onunload` в коде фронтенда



BLE

Библиотека Noble

```
var noble = require('noble');
var coordServiceUuid = '13333333333333333333333333333331';
var coordCharUuid = '000000000000000000000000000072646372';
noble.on('stateChange', function(state) {
  if (state==='poweredOn') {
    console.log("powered on");
    setInterval(function()
    {
      console.log("scanning");
      noble.startScanning([coordServiceUuid], true);
    }, 2000);
  }
  else {
    noble.stopScanning();
  }
})
```

Библиотека Noble

```
noble.on('discover', function(peripheral) {  
  console.log('found peripheral', peripheral.advertisement);  
  
  peripheral.connect(function(err) {  
    peripheral.discoverServices([coordServiceUuid], function(err, services) {  
      services.forEach(function(service) {  
        console.log('found service:', service.uuid);  
        connectionStatus = 1;  
        service.discoverCharacteristics([], function(err, characteristics) {  
          characteristics.forEach(function(characteristic) {  
            console.log('found characteristic:', characteristic.uuid);  
            if(coordCharUuid == characteristic.uuid) {  
              coordCharacteristic = characteristic;  
            }  
          })  
          if(coordCharacteristic) {  
            doCoords();  
          }  
        }  
      }  
    }  
  }  
})
```

...и огромная куча закрывающих скобок

Библиотека Noble

```
function doCoords()  
{  
  coordCharacteristic.on('read', function(data, isNotification) {  
    console.log(data.readUInt8(1));  
  });  
  coordCharacteristic.notify(true, function(error) {  
    console.log('notification on');  
    connectionStatus = 2;  
  });  
}
```