|  |
| --- |
| центр разработокD:\Users\daniil.barvitsky\Documents\Branding\VIAcode_Ai_new.emf |
| ООО «ИНТЕРСОФТ» |

Инструкция по установке и настройке

## Комплект поставки

Поставка состоит из .zip архива со следующими артефактами:

* Node.js приложение «Сервер»
* Три клиентских приложения:
  + Web-приложение «Монитор»
  + Web-приложение «Ведущий»
  + Android приложение «Игрок»
* Пример сценария игры

## Требования к системе

* Для запуска приложения «Игрок» необходим Web-браузер на любом устройстве с которого планируется игра
* Для запуска приложений «Монитор» и «Ведущий» необходим Web-браузер
* Для запуска приложения «Сервер» необходим Node.js актуальной версии

## Общая конфигурация

* Клиенты общаются с сервером по HTTP протоколу, при этом во все клиенты «зашито» доменное имя сервера
* localhost;соответственно DNS должен быть настроен так, чтобы это имя соответствовало IP адресу той машины, на которой будет развернут сервер.
* На машинах, на которых планируется запускать приложения «Сервер», «Монитор» и «Ведущий», необходимо установить **Node.js** (<http://nodejs.org/>).   
  Версия Node.js, под которую разрабатывалось и тестировалось приложение, – [v0.10.21 (x64)](http://nodejs.org/dist/v0.10.21/x64/).
* Самый простой сценарий предполагает установку и разворачивание приложений «Монитор», «Сервер» и «Ведущий» на одной машине.

## Сервер

«Сервер» представляет собой Node.js приложение, работающее на 3000 порту, для запуска которого необходимо проделать следующие шаги:

* Используя командную строку, перейти в папку сервера: **cd server**
* Запустить сервер: **node app.js**
* Убедиться, что сервер работает, открыв в браузере URI <http://localhst:3000/healthcheck>

В случае успешного запуска в браузере отобразится следующее:

"Wittendo Server is up and running"

## Монитор

Монитор представляет собой обычное браузерное HTML 5 приложение, запускать которое можно с использованием любого HTTP сервера, но мы тестировали и разрабатывали его опять же с помощью Node.js:

* **cd client-monitor\app**
* **node ..\scripts\web-server.js**
* <http://localhost:8000/index.html>

По умолчанию монитор разворачивается на порту 8000, но это может быть настроено в скрипте   
**client-monitor\scripts\web-server.js:**

var DEFAULT\_PORT = 8000;

## Ведущий

Запуск приложения «Ведущий» абсолютно аналогичен запуску «Монитора»:

* **cd client-host\app**
* **node ..\scripts\web-server.js**
* [http:// localhost:8001/index.html](http://wittendoserver:8001/index.html)

Настройка порта, опять же, производится в скрипте **client-host\scripts\web-server.js**.

## Игрок

«Игрок» представляет собой Web-приложение, открываемое на порту 8002.

Для того, что бы открыть и работать с ним необходимо:

1.Запустить сервер.

2.В файле index.html найти тег <script>,содержащий в себе строку:

” serverAddress = serverAddress || "http://YOU\_IP:3000" “

Здесь необходимо заменить YOU\_IP на валидный адрес , который можно найти выполнив:

2.1 WIN+R

2.2 cmd (затем выполнить)

2.3 В командной строке выполнить команду ipconfig

2.4 В строчке IP/v4 ,будет находится валидный YOU\_IP адрес.

3. Открыть в браузере http:// YOU\_IP:8002/client-player/assets/www/index.html.

## Сценарий игры и медиа файлы

Сценарий конкретной игры состоит из нескольких частей:

* Файл **server\content\scenario.json**, позволяющий задать сам сценарий игры. В поставку входит образец этого файла с комментариями о том, как им пользоваться. Обращаем внимание, что в файле сценария, который лежит на сервере, комментарии недопустимы.
* Медиа контент отображаемый на «Мониторе», - в **client-monitor\app\content** папки **audio**, **image** и **video**

Обращаем внимание, что видео и картинки растягиваются на весь экран монитора, поэтому они должны иметь разрешение, совпадающее с разрешением монитора.

* + Поддерживаемые видео кодеки – [Theora](http://ru.wikipedia.org/wiki/Theora), [H.264](http://ru.wikipedia.org/wiki/H.264), [WebM](http://en.wikipedia.org/wiki/WebM_Project).
  + Поддерживаемые форматы аудио – MP3.
  + Поддерживаемые форматы картинок – GIF, JPEG, PNG.

При старте «Монитора» проверяется список медиа файлов, используемых в текущем сценарии, с отображением недостающих. Однако есть возможность продолжить игру и без недостающих медиа файлов.

Запросить массив медиа файлов для проверки можно по ссылке <http://localhost:3000/api/game/medias>.

* Звуковые файлы, используемые в игре, **-** в **client-monitor\app\content** папка **sound**

Используются следующие аудио файлы:

**Раунд с простым вопросом:**

1. После нажатия «Начать раунд» – **start\_round.mp3** (ограничения на продолжительность нет)
2. После нажатия «Показать вопрос» – **before\_showQuestion.mp3** (ограничения на продолжительность нет)
3. После нажатия «Показать варианты»

* Перед запуском таймера – **before\_startTimer.mp3** (рекомендованная длина 1 секунда)
* Тикание таймера до 5 секунд – **runningTimer.mp3** (рекомендованная длина 1 секунда)
* Тикание таймера от 5 до 0 секунд – **runningTimerLastSeconds.mp3** (рекомендованная длина 1 секунда)
* После остановки таймера – **after\_stopTimer.mp3** (рекомендованная длина 1 секунда)

1. После нажатия «Показать правильный ответ» – **before\_rightAnswer.mp3** (ограничения на продолжительность нет)

**Раунд «Дум-дум экспресс»:**

1. После нажатия «Начать раунд» – **start\_round.mp3** (ограничения на продолжительность нет)
2. После нажатия «Запустить Дум-Дум Экспресс»

* Перед отображением каждого вопроса – **start\_express\_N.mp3** (где N – номер вопроса в экпрессе) (рекомендованная длина 1 секунда)
* После остановки таймера – **after\_stopTimer.mp3** (рекомендованная длина 1 секунда)

1. После нажатия «Показать правильный ответ» – **before\_rightAnswer.mp3** (ограничения на продолжительность нет)

## Блокирование команды

Между раундами ведущий может вывести команду, нарушившую правила, на несколько раундов из текущей игры. Для этого на вкладке ИТОГИ напротив нужной команды нужно кликнуть Off (Блокировать команду) или On (Вернуть команду в игру). При этом на планшете блокируется вкладка ИГРА (остальные вкладки будут доступны). Далее при необходимости предполагается изъять планшет у команды.

## Мониторинг доступности планшетов игроков

В приложении «Ведущего» предусмотрен мониторинг статусов доступности планшетов игроков. Если планшет доступен, рядом с названием команды будет отображаться зеленая галочка, в противном случае – замигает красный треугольник с восклицательным знаком.

Что делать, если ведущий видит, что одна или несколько команд стали недоступны:

1. Временно остановить игру: не продолжать игру на стороне приложения «Ведущего».
2. Не закрывать клиентские приложения на планшетах, иначе текущий прогресс игры будет утерян.
3. Проверить доступность сети Wifi на планшетах команд, которые столкнулись с проблемой.
4. После восстановления сети на планшетах попробовать попереключаться по вкладкам приложения, тем самым сгенерировав запросы к серверу. Возможно придется немного подождать.
5. После восстановления доступности команд игру можно продолжить.

## Рестарт игры

Перезапуск игры можно осуществить одним из двух способов:

1.

Остановить запущенный сервер и перезапустить командами:

* **cd server**
* **node app.js**

2.

Перейдя по ссылке [http:// localhost:3000](http://wittendoserver:3000) и кликнув «Перезапустить игру».

После рестарта сервера перезапускать приложения «Монитор» и «Ведущий» не требуется. Достаточно вернуться на их главную страницу index.html.

Для начала новой игры осталось только перезапустить Android клиенты «Игрок» на планшетах.