

Угадайка

Сегодня мы попробуем создать простую и логичную игру, в которую сможет сыграть любой желающий через интернет.

Игра называться "Угадай число". Цель игры проста и понятна из названия, программа должна отгадать, какое число загадал пользователь. Программа может только спросить:

"Число, которое было загадано, это \$number? " и пользователь может ответить программе:

1. Да, это моё число.
2. Нет, моё число меньше.
3. Нет, моё число больше.

И в зависимости от этого будет меняться последующий вопрос или же программа угадала. Для этой цели нам понадобится 3 страницы

1. Приветствие и ознакомление с игрой.
2. Угадывание числа.
3. Вывод числа, которое загадал пользователь.

Теперь рассмотрим каждую страницу подробнее

Страница первая, приветствие и ознакомление с игрой.

На первой странице будет заглавная надпись с названием игры, приветствие игрока, объяснение правил игры и что нужно делать пользователю. После всего текста внизу должна быть кнопка, при нажатии на которую, пользователь переходит на главную (вторую) страницу, вместе с этим передаются параметры самого большого (max) и самого маленького (min) числа.

Теперь подробнее об этом.

Создание страницы

Для начала, создадим саму страницу:

```
<?php

require 'vendor/autoload.php';

use \atk4\ui\Header;

//Создание "App" объекта уже было разобрана на предидущих уроках.
$app = new \atk4\ui\App('Угадайка!');

$app->initLayout('Centered');
```

Текстовое оформление

После этого необходимо вывести все заголовки:

```
// Создаём заголовок с надписью "Инструкция".
$app->add(['Header', 'Инструкция']);

// Добавим две компоненты и настроим их:
// Вывод правил обычным текстом.
$text = $app->add(['Text']);
$text->addParagraph('Эта игра демонстрирует, как использовать Links, Buttons, Headers, Text в Agile Toolkit.');
```

\$text->addParagraph("Для игры Вам необходимо загадать любое целое число от 1 до 100 включительно. После того, как Вы загадали число, нажмите кнопку 'Начать игру!' ");

Создание кнопки

Последним пунктом для этой страницы является кнопка, которая переводит пользователя на следующую страницу, и передаёт параметры для \$max и \$min:

```
// Создаём кнопку, с названием "Начать игру!" и добавляем к ней справа кнопку со смайликом
$button = $app->layout->add(['Button', "Начать игру!", 'iconRight'=>'smile']);

// Атрибут "primary" ставиться кнопке к которой мы хотим привлечь внимание пользователя. Это хороший тон программирования, для того, что бы привлечь к этой кнопки особое внимание. Не следует делать все кнопки с атрибутом "primary" потому как внимание пользователя будет утеряно.
$button->set(['primary'=>true]);

// И для полноценной работы кнопки, указываем куда она будет вести // и какие параметры передаёт.
// Передача параметров тоже уже была разобрана на уроке.
$button->link(['main', 'max'=>100, 'min'=>0]);
```

Вот как это всё должно выглядеть:



Добро пожаловать в игру!

Инструкция

Эта игра демонстрирует, как использовать Links, Buttons, Headers, Text в Agile Toolkit.

Для игры Вам необходимо загадать любое целое число от 1 до 100 включительно. После того, как Вы загадали число, нажмите кнопку 'Начать игру!'

Начать игру!



Страница вторая, угадывание числа.

Вторая страница создаётся точно таким же путём, как и первая, следовательно у неё есть такая же заглавная надпись:

```
<?php

require 'vendor/autoload.php';

use \atk4\ui\Header;

$app = new \atk4\ui\App( 'Угадайка!' );
$app->initLayout( 'Centered' );
```

После чего надо считать переданных на эту страницу 2 переменные **max** и **min**. Имея их и зная к какому числу мы будем задавать вопрос, мы вычисляем само число **number**. Вот как это происходит:

```
// Считываем переменную max.
$max = $_GET[ 'max' ];

// Считываем переменную min.
$min = $_GET[ 'min' ];

// Через среднее арифметическое и округление вычисляем number.
$number = round( ( $max + $min ) / 2 );
```

Далее пишем текст с вопросом о том, является ли **number** тем, что загадал пользователь:

```
$app->add( new Header( [ 'Ваше число это ' . $number . ' ?', 'size' => 1 ] ) );
```

Далее под надпись с вопросом создаём 3 кнопки с вариантами ответов:

1. **Да, это моё число**
2. **Нет, моё число меньше**
3. **Нет, моё число больше**

Если пользователь нажимает на кнопку **"Да, это моё число"**, то необходимо перейти на третью страницу и ей передать в виде параметра, какое же число было загадано пользователем (number).

```
// Создаём кнопку, даём ей название и "рисую" иконку.  
$button = $app->layout->add(['Button', "Да, это моё число!", 'icon'=>'empty  
star']);  
  
// Так как эта кнопка для нас обозначает победу,  
// то её мы выделяем по отношению к другим.  
$button->set(['primary'=>true]);  
  
// Даём ссылку на следующую, третью, страницу и параметры, какое число было  
загадано.  
$button->link(['win', 'number'=>$number]);
```

Если пользователь нажал на кнопку **"Нет, моё число меньше"**, то минимум передаётся так же, а вместо максимума передаётся предполагаемое на данный момент число.

```
// Создаём кнопку с названием "Нет, моё число меньше"  
$button = $app->layout->add(['Button', 'Нет, моё число  
меньше.', 'icon'=>'arrow down']);  
  
// Прикрепляем ссылку на эту же страницу и изменяем параметр max.  
// Так как число которое было загадано, оказалось  
// меньше предполагаемого, то мы сдвигаемся влево  
// по числовой оси и max становится на место number.  
$button->link(['main', 'max'=>$number, 'min'=>$min]);
```

Если пользователь нажал на кнопку **"Нет, моё число больше"**, то вместо минимума передаётся предполагаемое на данный момент число, а максимум передаётся так же.

```
// Создаём кнопку с надписью "Нет, моё число больше".  
$button = $app->layout->add(['Button', 'Нет, моё число  
больше.', 'icon'=>'arrow up']);  
  
// Создаём ссылку на эту же страницу, но изменяем параметр min.  
// Смещение по числовой оси произошло вправо, по этому  
// min становится number, а max не изменятся.  
$button->link(['main', 'max'=>$max, 'min'=>$number]);
```

Где бы пользователь не находился в нашей программе, правильный стиль это если у него есть возможность вернуться в начало программы. Это не следует акцентировать, поэтому мы будем пользоваться обычной стандартной ссылкой на первую страницу. Точно такая же часть кода будет и на третьей странице. По этому нужно сделать кнопку "Сыграть ещё раз".

Если пользователь нажал на кнопку **"Сыграть ещё раз"**, то происходит переход на первую страницу.

```
// Создаём кнопку для перезапуска игры с названием "Сыграть ещё раз".
$button = $app->layout->add([ 'Button', 'Сыграть ещё
раз.', 'icon'=>'refresh' ] );

// Ссылку создаем на эту же страницу
// с изначальными параметрами.
$button->link([ 'index' ] );
```

Теперь самая главная часть готова. Осталось только создать страницу, где программа показывает число, которое загадал пользователь.

Вот, что получается в итоге:



Ваше число это 50 ?

☆ Да, это моё число!

↓ Нет, моё число меньше.

↑ Нет, моё число больше.

↺ Сыграть ещё раз.

Страница третья, вывод числа, которое загадал пользователь.

Начало третьей странице точно такое же — её создание:

```
<?php

require 'vendor/autoload.php';

use \atk4\ui\Header;

$app = new \atk4\ui\App( 'Угадайка!' );
$app->initLayout( 'Centered' );
```

На третьей странице есть главная надпись, которая красиво оформлена. Перед её оформлением необходимо получить значение (number) через \$_GET. В ней говорится, что загаданное число это (number):

```
// Считываем полученное число, которое мы угадали.  
$number = $_GET['number'];  
  
// Вывод number на экран.  
$app->add(['header', ' Я выиграл, Ваше загаданное число, это '.$number.'  
!']);
```

Ниже менее заметна кнопка, предлагающая сыграть ещё раз. По ссылке и передачи значений она идентична с кнопкой «Сыграть ещё раз» на второй странице.

```
// Создаём саму кнопку с соответствующим названием  
// и прикрепляем рисунок перевернутых стрелок  
$button = $app->layout->add(['Button', 'Сыграть ещё  
раз.', 'iconRight'=>'refresh']);  
  
// Что бы кнопка была не заметна, делаем её маленькой (small).  
$button->set(['size small'=>true]);  
  
// Добавляем ссылку, которая ведёт на вторую страницу  
// с изначальными данными.  
$button->link(['main', 'max'=>100, 'min'=>0]);
```

Так будет выглядеть третья страница, если загадывали число 83:



Добро пожаловать в игру!

Я выиграл, Ваше загаданное число, это 83 !

Сыграть ещё раз.

