**Научно-практическая конференция «От идеи к изобретению»**

**Умное зеркало**

Работа выполнена обучающейся 10 класса

МБОУ «Междуреченская СШ №6» Пинежский район

Омельченко Аделиной Юрьевной

**г. Северодвинск, 2020**

**ОГЛАВЛЕНИЕ**

Введение 3

Описание работы среды Magic Mirror 4

Описание аппаратного комплекса системы для вывода информации 5

Описание модулей для вывода информации 7

Заключение 9

Библиографический список 10

Приложение 1. Фотография модели 11

Приложение 2. Часть кода модуля CurrentWeather 12

Приложение 3. Часть кода модулей Calendar, NewsFeed 13

Приложение 4. Часть кода модуля Compliments 14

**Введение**

Каждый день мы сталкиваемся с системами для вывода информации: монитор ПК, принтер, смартфон, мультимедийный проектор, колонки и наушники. Мы узнаём новости по радио, проверяем прогноз погоды по телевизору, используем компьютер для получения нужной информации, для общения, с помощью принтера мы можем передавать информацию с электронного носителя в печатный вид. Существует 8 основных типов дисплеев, используемых человеком в быту: жидкокристаллические, LCD, TFT, IPS, TN, LED, OLED, плазменная панель [[1]](https://monitor4ik.com/stati/vsyo-pro-matritsy-monitora-tn-ips-pls-va-mva-oled/). Системы для вывода информации актуальны и в наши дни. Системы для вывода информации оказывают положительных эффект в развитии детей при правильном их использовании. Компьютер сегодня является средством для общения, для людей с ограниченными возможностями это единственный способ не только общения, но и самореализации.

Цель - создать программно-аппаратный комплекс для вывода актуальной информации.

Объект исследования - программно-аппаратный комплекс вывода информации.

В соответствии с данной целью были поставлены следующие задачи:

1. Описать работу среды для вывода изображения на дисплей;
2. Создать и описать аппаратный комплекс для вывода актуальной информации для человека;
3. Описать программу для вывода информации на экран.

Методы исследования: теоретическое и практическое моделирование, эксперимент.

**Описание работы среды для вывода изображения на дисплей**

Существует множество различных сред для вывода необходимой информации на дисплей**.** В операционной системе Linux наиболее удобной нам представляется среда Magic Mirror **-** это открытый исходный код, бесплатный и поддерживаемый большой группой энтузиастов, созданный в 2016 году [[2]](https://magicmirror.builders/). Ядро MagicMirror содержит API, который позволяет сторонним разработчикам создавать дополнительные модули.

API (программный интерфейс приложения, интерфейс прикладного программирования) — описание способов (набор классов, процедур, функций, структур или констант), которыми одна компьютерная программа может взаимодействовать с другой программой. Обычно входит в описание какого-либо интернет-протокола, программного каркаса или стандарта вызовов функций операционной системы. Часто реализуется отдельной программной библиотекой или сервисом операционной системы [[3]](https://ru.wikipedia.org/wiki/API).

Пользователь может устанавливать любые модули для вывода необходимой информации или же создавать их сам. Для форматирования используются следующие языки программирования: HTML, CSS и Javascript. помимо стандартных модулей выбираются дополнительные по усмотрению пользователя.

Все модули загружаются в папку modules. Модули по умолчанию группируются в папке modules/default.

**Описание аппаратного комплекса системы для вывода информации**

Для дальнейшего эксперимента использовался микрокомпьютер с установленной операционной системой Linux Raspbian [[4]](https://ru.wikipedia.org/wiki/Raspbian). Ядро Linux и основные компоненты, из которых состоит система, и множество программ распространяются с открытым исходным кодом бесплатно. Linux очень нетребователен к ресурсам. Linux поддерживает практически все основные языки программирования (Python, C / C ++, Java, Perl, Ruby и т.д .). Доступность исходных текстов Linux дает возможность использовать и модифицировать код по своему желанию. Linux имеет очень гибкие и обширные настройки. Linux очень надежная и стабильная система.

Для проведения эксперимента был использован одноплатный микрокомпьютер RaspberryPi 3. Он имеет 64-х битную платформу ARMv8-A Cortex-A53 с тактовой частотой 1.2 ГГц. Микрокомпьютер поддерживает работу беспроводных мышей и клавиатур. На плате присутствуют два модуля USB 2.0, microUSB, аудио разъем 3.5 мм и HDMI. Поддерживается 1Гб оперативной памяти. На плате имеется контроллер WLAN и Bluetooth модуль. Таким образом, компьютер имеет возможность получать информацию из сети Интернет, поддерживает различные протоколы передачи данных и имеет возможность подключения датчиков через встроенные цифровые порты. Выбранный микрокомпьютер компактен, он потребляет малое количество энергии и требует активной системы охлаждения. Для создания полупрозрачного зеркала использовалась тонкая зеркальная плёнка, наклеенная на кусок стекла, равный размеру экрана телевизора, который и используется для вывода необходимой пользователю информации. Устройство помещено в деревянный корпус с блоком питания 5 В. [[5]](https://micro-pi.ru/raspberry-pi-3-model-b-rpi-bcm2837/) Для экономии электроэнергии на корпус помещён модуль датчика движения на базе чипа KU5529. Тип датчика – радиолокационный, может работать за стеклом. Дальность обнаружения до 8 метров. При обнаружении объёма перед зеркалом датчик посылает логическую единицу в виде сигнала 3,3 В, тем самым включая реле, управляющее напрямую светодиодным блоком подсветки внутри телевизора. Если движение отсутствует в течение более 40 секунд, реле выключает подсветку, но микроконтроллер продолжает работу.

**Описание модулей для вывода информации**

На наше «умное» зеркало будет выводиться необходимая пользователю информация. Это данные о состоянии погоды, календарь, время, последние новости из выбранных RSS каналов. Эмоциональную составляющую добавляет модуль вывода комплиментов, определяемых временем суток.

Модуль currentweather отображает текущую погоду, включая скорость ветра, время захода или восхода солнца, температуру и значок для отображения текущих условий. Мы настроили следующие свойства: Чтобы получить данные, которые мы отобразили в зеркале, требуется использование некоторых открытых API и фидов. При регистрации на сайте [https://openweathermap.org/price](https://openweathermap.org/price%D1%8A) предоставляется бесплатный уникальный идентификатор (API), с помощью которого мы сможем получать актуальные данные о погоде (Приложение 2). Регистрация нужна для получения идентифицирующей пользователя строки App Id, состоящей из набора букв и цифр (похоже — только из шестнадцатеричных цифр) Такого вида:«6d8e495ca73d5bbc1d6bf8ebd52c4». После регистрации нужно зайти в личный кабинет и взять App Id, который там называется «API key». Для населённого пункта Карпогоры код идентификатора был найден на представленном выше сайте. [6]

Модуль compliments отображает случайный комплимент. Он содержит объект с тремя массивами: morning (с 06:00 до 12:00), afternoon(с 12:00 до 18:00) и evening (с 18:00 до 24:00). В зависимости от времени суток, комплименты будут выбраны из одного из этих массивов. Массивы содержат один или несколько комплиментов. Утром отражаются следующие комплименты: «Доброе утро!», «Чудесного дня!», «Прекрасное утро!». Днём отображаются следующие комплименты: «Прекрасный день!», «Жизнь хороша!», «Ты молодец!». Вечером отображаются следующие комплименты: «Приятных сновидений!», «Сладких снов!», «Доброй ночи!». В любое время суток отображается комплимент «Прекрасно выглядишь!». (Приложение 4)

Модуль newsfeed отображает заголовки новостей на основе RSS-канала. RSS — семейство XML-форматов, предназначенных для описания лент новостей, анонсов статей, изменений в блогах и т. п. Информация из различных источников, представленная в формате RSS, может быть собрана, обработана и представлена пользователю в удобном для него виде специальными программами-агрегаторами или онлайн-сервисами, такими как: NewsAlloy, FeedBucket и другими. Обычно с помощью RSS 2.0 даётся краткое описание новой информации, появившейся на сайте, и ссылка на её полную версию. Интернет-ресурс в формате RSS называется RSS-каналом, RSS-лентой или RSS-фидом. [[7]](https://ru.wikipedia.org/wiki/RSS) В качестве RSS-ленты мы использовали сервис Google новости. [[8]](https://news.google.com/?hl=ru&gl=RU&ceid=RU:ru) Для отображения актуальных для пользователя новостей настраиваются сайты, интересные пользователю. Прокрутка заголовков новостей происходит по времени, но также может управляться отправкой определенных уведомлений Ленты Новостей в модуль. (Приложение 3)

Модуль calendar отображает все дни года с указанием различных других сведений. Например, событий или важных дат. За основу был взят сервис Google календарь. (Приложение 3)

Модуль clock отображает текущую дату и время. Информация будет обновляться в реальном времени. Мы настроили следующие свойства: 12 или 24-часовой формат, отображение секунд, отображение цифровых часов, аналоговых часов или обоих вместе.

**Заключение**

В ходе проделанной работы была изучена проблема создания программно-аппаратного комплекса системы для вывода информации, создана действующая модель. Созданная модель имеет практическое применение. В дальнейшем она может быть усовершенствована путем добавления и создания новых модулей для программы, например: измерение температуры в помещении, проигрывание видео материалов, добавление сенсорного экрана, возможность прослушивания музыки, использования датчиков влажности и атмосферного давления.

**Библиографический список**

1. Типы дисплеев [Электронный ресурс] URL: <https://monitor4ik.com/stati/vsyo-pro-matritsy-monitora-tn-ips-pls-va-mva-oled/>
2. MagicMirror [Электронный ресурс] URL: <https://magicmirror.builders>
3. Информация о API [Электронный ресурс] URL: https://ru.wikipedia.org/wiki/API
4. Linux [Электронный ресурс] URL: https://ru.wikipedia.org/wiki/Raspbian
5. Информация о Raspberry Pi 3 [Электронный ресурс] URL:<https://micro-pi.ru/raspberry-pi-3-model-b-rpi-bcm2837/>
6. API погоды населённого пункта [Электронный ресурс] URL:<https://habr.com/ru/post/315264/>
7. RSS-лента [Электронный ресурс] URL: <https://ru.wikipedia.org/wiki/RSS>
8. Google новости [Электронный ресурс] URL: https://news.google.com/?hl=ru&gl=RU&ceid=RU:ru

ПРИЛОЖЕНИЕ 1

Фотография модели



Модель с датчиком приближения, в деревянном корпусе с полупрозрачным зеркалом. На зеркале модули «Календарь», «Комплименты», «Новости» и «Погода»

Приложение 2

Часть кода модуля «CurrentWeather»

apiBase: "https://api.openweathermap.org/data/",

 forecastEndpoint: "forecast/daily",

 appendLocationNameToHeader: true,

 calendarClass: "calendar",

 tableClass: "small",

 roundTemp: false,

 iconTable: {

 "01d": "wi-day-sunny",

 "02d": "wi-day-cloudy",

 "03d": "wi-cloudy",

 "04d": "wi-cloudy-windy",

 "09d": "wi-showers",

 "10d": "wi-rain",

 "11d": "wi-thunderstorm",

 "13d": "wi-snow",

 "50d": "wi-fog",

 "01n": "wi-night-clear",

 "02n": "wi-night-cloudy",

 "03n": "wi-night-cloudy",

 "04n": "wi-night-cloudy",

 "09n": "wi-night-showers",

 "10n": "wi-night-rain",

 "11n": "wi-night-thunderstorm",

 "13n": "wi-night-snow",

 "50n": "wi-night-alt-cloudy-windy"

 },

 },

 firstEvent: false,

 fetchedLocationName: "14761",

Приложение 3

Часть кода модулей «NewsFeed», «Calendar»

 NEWSFEED

modules: [

 {

 module: "newsfeed",

 position: "bottom\_bar", center regions.

 config: {

 feeds: [

 {

 title: "Google Feed",

 url: «<https://news.google.com/?hl=ru&gl=RU&ceid=RU:ru>»,

 },

]

CALENDAR

calendars: [

 {

 symbol: "calendar",

 url: "https://calendar.google.com/calendar/",

 },

 ],

 titleReplace: {

 "Сегодня ": "",

 "'Завтра": ""

 },

 broadcastEvents: true,

 excludedEvents: [],

 sliceMultiDayEvents: false,

 broadcastPastEvents: false,

 nextDaysRelative: false

Приложение 4

Часть кода модуля Compliments

Defaults: {

 Compliments: {

 Anytime: [

 «Прекрасно выглядишь!»

 ],

 Morning: [

 «Доброе утро!»,

 «Чудесного дня!»,

 «Прекрасное утро!»

 ],

 afternoon: [

 «Прекрасный день!»,

 «Жизнь хороша!»,

 «Ты молодец!»

 ],

 Evening: [

 «Приятных сновидений!»,

 «Сладких снов!»,

 «Доброй ночи!»

 ]

},

updateInterval: 30000,