# ОГЛАВЛЕНИЕ

[ОГЛАВЛЕНИЕ 3](#_Toc2253112)

[1. ИСХОДНЫЕ ДАННЫЕ 4](#_Toc2253113)

[1.1. Постановка задачи 4](#_Toc2253114)

[2. РЕАЛИЗАЦИЯ 6](#_Toc2253115)

[2.1. Портал Azure 6](#_Toc2253116)

[2.2. Подготовка графа 6](#_Toc2253117)

[3. ТЕСТИРОВАНИЕ 9](#_Toc2253118)

[4. ЗАКЛЮЧЕНИЕ 10](#_Toc2253119)

[5. СПИСОК ЛИТЕРАТУРЫ 11](#_Toc2253120)

# ИСХОДНЫЕ ДАННЫЕ

# Постановка задачи

Концепция Интернета вещей (IoT) не нова, однако особенную популярность она приобрела именно в последние годы. Данная концепция представляет собой вычислительную сеть физических предметов, оснащенных встроенными технологиями для взаимодействия друг с другом или с внешней средой [1]. Общий мировой объем капиталовложений в IoT в 2018 году составил 646 миллиардов долларов США. Прогноз на 2019 год — 745 миллиардов долларов США, на 2022 — более 1 триллиона долларов [2].

Развитие распределенной сетевой инфраструктуры в автоматизированных системах управления технологическим процессом привело к появлению Промышленного интернета вещей (IIoT) — концепции взаимосвязи датчиков, приборов и других устройств, объединенных в сеть с промышленными приложениями компьютеров, включая, помимо прочего, управление производством и энергопотреблением. Такая связь позволяет собирать данные и анализировать их, что потенциально способствует повышению производительности и эффективности, а также другим экономическим преимуществам [3]. Одним из преимуществ внедрения этого подхода является возможность создания цифрового двойника разрабатываемой системы (Digital Twin). Цифровой двойник — это иерархическая система математических моделей, вычислительных методов и программного обеспечения, которая обеспечивает синхронизацию между состоянием реально существующего процесса или системы и сопутствующей виртуальной копией. Эта концепция рассмотрена в статьях [4, 7].

Благодаря развитию облачных технологий появилась возможность создать инфраструктуру хранения данных, способную поддерживать Интернет вещей. Публичные облачные платформы предоставляют множество решений для цифрового преобразования бизнеса. Гибкость и автоматизированность полученной среды достигается за счет использования прикладных программных интерфейсов (API). Это позволяет различным устройствам и системам взаимодействовать между собой, даже если они работают на основе разных стандартов и протоколов [5].

Применение облачных технологий может упростить обработку сверхбольших баз данных, генерируемых устройствами Интернета вещей. Это показывает необходимость изучения инструментов облачных платформ для разработки приложений, обрабатывающих данные с устройств IoT. Одним из возможных решений являются облачная платформа Microsoft Azure. Ключевым преимуществом данной платформы является широкий набор инструментов, в частности, для решений Интернета вещей и цифровых двойников. Azure Digital Twins от Microsoft. Azure Digital Twins представляет собой службу Интернета вещей, с помощью которой можно создать комплексные модели физического окружения. С ее помощью можно создавать пространственные интеллектуальные графы для моделирования связей и взаимодействий между людьми, пространствами и устройствами. Azure Digital Twins позволяет запрашивать данные из физического пространства, а не из многих разрозненных датчиков [6].

Целью данной работы является разработка технологии создания цифровых двойников на основе ресурсов облачной вычислительной платформы Microsoft Azure. Для разработки технологии выделен следующий перечень задач:

1. Провести обзор научной литературы и существующих решений интернета вещей и цифровых двойников
2. Проанализировать технологии, предоставляемые облачными вычислительными платформами, для создания цифровых двойников
3. Реализовать приложение для обработки данных, генерируемых устройствами Интернета вещей и провести его тестирование

# РЕАЛИЗАЦИЯ

В работе представлена реализация приложения для представления офисного здания в виде пространственного графа и создания его цифрового двойника. С помощью технологии Azure Digital Twins можно связать несколько датчиков со своей средой. Так, в разработанном приложении смоделированный датчик углекислого газа позволяет выяснить, оптимальный ли в помещении воздух. Ввиду акцентирования данной работы на разработке именно технологии, были использованы случайные данные, близкие по своим значениям к настоящим.

# Портал Azure

На портале Microsoft Azure был создан ресурс, представляющие экземпляр Azure Digital Twins, а также произведена регистрация приложения в облачной службе управления удостоверениями и доступом Azure Active Directory (AD), благодаря чему у приложения есть доступ к созданному экземпляру Digital Twins. Azure AD помогает получать доступ к внешним ресурсам, таким как портал Azure, а также предоставляет API, с помощью которых можно разрабатывать персонализированные интерфейсы приложения.

# Подготовка графа

Сборка приложения начинается с заполнения JSON-файла конфигурации appSettings.json, который должен содержать следующие переменные:

1. ClientID – идентификатор созданного приложения Active Directory.
2. Tenant – идентификатор созданного каталога клиента Active Directory.
3. BaseURL – URL-адрес API управления экземпляром Digital Twins.

На следующем шаге подготавливается пространственный граф Digital Twins со следующими элементами, количество которых зависит от конфигурации:

1. Пространства;
2. Устройства;
3. Датчики;
4. Пользовательские функции;
5. Назначенные роли.

Пространственный граф подготовлен с помощью YAML-файла, описывающего его структуру. На данном этапе также будет создан Центр Интернета вещей. Далее, экземпляр Digital Twins переходит в состояние Running, после чего можно осуществлять отправку данных датчиков, чтобы считать и впоследствии обработать их. На рис. 1 показано визуализация функции сбора данных с датчиков, включающее непосредственно значение, дату, время, свойства датчиков, а также уникальные ключи запросов в формате GUID (Globally Unique Identifier). Часть реализации данной функции показана на рис. 3.

|  |
| --- |
| 26.02.2019 0:22:31> Sending message: {"SensorValue":"true"} Properties: { 'DigitalTwins-Telemetry': '1.0','DigitalTwins-SensorHardwareId': 'SAMPLE\_SENSOR\_MOTION','CreationTimeUtc': '2019-02-25T19:22:31.5018987Z','x-ms-client-request-id': 'ce34774e-316e-45a7-baef-74e2659b195a', }  26.02.2019 0:22:31> Sending message: {"SensorValue":"1035"} Properties: { 'DigitalTwins-Telemetry': '1.0','DigitalTwins-SensorHardwareId': 'SAMPLE\_SENSOR\_CARBONDIOXIDE','CreationTimeUtc': '2019-02-25T19:22:31.7026635Z','x-ms-client-request-id': 'a06187ae-78a8-42de-9d9b-e8cea99be9d1', }  26.02.2019 0:22:32> Sending message: {"SensorValue":"82"} Properties: { 'DigitalTwins-Telemetry': '1.0','DigitalTwins-SensorHardwareId': 'SAMLE\_TEMPERATURE\_SENSOR','CreationTimeUtc': '2019-02-25T19:22:32.0018568Z','x-ms-client-request-id': '883076f0-61d3-4741-b274-8d8ffbac932f', } |
| Рис. 1. Вывод данных с датчиков |

Для реализации приложения был использован ООП язык программирования C#, а также специальные библиотеки для работы с инструментами Microsoft Azure, такие как Microsoft.Azure.Devices.Client, Microsoft.Azure.DigitalTwins и другие.

Каждые 5 секунд датчики имитируют и отправляют значения случайных данных для трех измерений — движения, углекислого газа и температуры. Свободные пространства со свежим воздухом определяются отсутствием людей в помещении, а также уровнем углекислого газа ниже 1000 ppm (частей на миллион). Если условие не выполняется, то пространство считается недоступным или качество воздуха в нем принимается за плохое.

В другой командной строке в режиме реального времени считывается граф для поиска свободных помещений со свежим воздухом на основе случайных смоделированных данных. В зависимости от того, какие данные датчиков были отправлены последними, в этой строке в режиме реального времени отобразится одно из следующих условий:

* Свободные помещения со свежим воздухом;
* Занятая комната или комната с плохим качеством воздуха.

На рис. 2 показана визуализация функции определения качества заданного помещения, содержащая название помещения, его идентификатор в формате GUID, время и дату, а также значение. На данном рисунке показаны оба возможных варианта определенного на основе данных, поступающих в реальном времени, качества помещения.

|  |
| --- |
| Name: Focus Room A1  Id: e212dc1b-501f-4f69-96d9-78f12ab7aab5  Timestamp: 2019-02-25T19:22:15.1562462Z  Value: Room is available and air is fresh  Name: Focus Room A1  Id: e212dc1b-501f-4f69-96d9-78f12ab7aab5  Timestamp: 2019-02-25T19:22:20.0876775Z  Value: Room is not available or air quality is poor |
| Рис. 2. Определение качества помещения |
| private static IConfigurationSection settings;  static void Main(string[] args)  {  settings = new ConfigurationBuilder()  .SetBasePath(Directory.GetCurrentDirectory())  .AddJsonFile("appsettings.json")  .Build()  .GetSection("Settings");  try  {  DeviceClient deviceClient = DeviceClient.CreateFromConnectionString(settings["DeviceConnectionString"]);  if (deviceClient == null)  {  Console.WriteLine("ERROR: Failed to create DeviceClient!");  return;  }  SendEvent(deviceClient).Wait();  }  } |
| Рис. 3. Реализация подключения к клиенту |

# ТЕСТИРОВАНИЕ

Для проверки работоспособности системы было проведено функциональное тестирование, подтверждающее правильность полученных значений качества помещений. Результаты тестирования представлены на табл. 1.

Табл. 1. Результаты функционального тестирования

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **№** | **Название теста** | **Шаги** | **Ожидаемый результат** | **Тест пройден?** |
| 1 | Помещение недоступно | 1. Выявляется показание датчика с уровнем углекислого газа выше допустимой нормы или наличием людей в помещении 2. Производится анализ полученного в реальном времени значения | На экране должно отобразиться следующее сообщение: Room is not available or air quality is poor. | Да |
| 2 | Помещение доступно | 1. Выявляется показание датчика с уровнем углекислого газа ниже допустимой нормы и отсутствием людей в помещении 2. Производится анализ полученного в реальном времени значения | На экране должно отобразиться следующее сообщение: Room is available and air is fresh. | Да |

# ЗАКЛЮЧЕНИЕ

В рамках данной работы был произведен обзор научной литературы и существующих решений Интернета вещей и цифровых двойников, проанализированы технологии, предоставляемые облачными вычислительными платформами, для создания цифровых двойников, а также реализовано и протестировано приложение для обработки данных, генерируемых устройствами Интернета вещей.

# СПИСОК ЛИТЕРАТУРЫ

1. Internet Of Things. Gartner IT glossary. Gartner (5 May 2012). [Электронный ресурс] URL: https://www.gartner.com/it-glossary/internet-of-things/ (дата обращения: 04.02.2019).
2. IDC Forecasts Worldwide Spending on the Internet of Things in 2019. [Электронный ресурс] URL: <https://www.idc.com/getdoc.jsp?containerId=prUS44596319/> (дата обращения: 04.02.2019).
3. Boyes, Hugh; Hallaq, Bil; Cunningham, Joe; Watson, Tim. The industrial internet of things (IIoT): An analysis framework. Computers in Industry, October, 2018. Vol. 101. – P. 1–12.
4. K. Borodulin, G. Radchenko, A. Shestakov, L. Sokolinsky, A. Tchernykh, R. Prodan, "Towards Digital Twins Cloud Platform: Microservices and Computational Workflows to Rule a Smart Factory", Proc. the10th Int. Conf. Util. Cloud Comput. - UCC '17, pp. 209–210, December 2017.
5. Интернет вещей: Будущее уже здесь. С. Грингард. М.: Альпина Паблишер, 2016. – 188 с.
6. Документация по Microsoft Azure. Micrisift Docs. [Электронный ресурс] URL: <https://docs.microsoft.com/ru-ru/azure/> (дата обращения: 04.02.2019).
7. G. Radchenko, A. Alaasam, A. Tchernykh, “Micro-Workflows: Kafka and Kepler fusion to support Digital Twins of Industrial Processes”, IEEE/ACM Int. Conf. Util. Cloud Comput. – UCC '18, pp. 83-88, December 2018.