

**ВЫПУСКНАЯ КВАЛИФИКАЦИОННАЯ РАБОТА  
МАГИСТЕРСКАЯ ДИССЕРТАЦИЯ**

**Тема** Разработка среды освоения когнитивных сервисов в виртуальной компьютерной лаборатории

**ФИО студента** Логинов Владимир Сергеевич

**Группа** 6014 **Направление подготовки** 27.04.03 - Системный анализ и управление

**Профильная направленность образовательной программы:** Интеллектуальные системы обработки больших данных

**Выпускающая кафедра** Системного анализа и управления

**Руководитель работы** \_\_\_\_\_ /доц., к.т.н. Белов М. А./

**Консультант(ы)**

\_\_\_\_\_/ \_\_\_\_\_/  
\_\_\_\_\_/ \_\_\_\_\_/  
\_\_\_\_\_/ \_\_\_\_\_/

**Рецензент** \_\_\_\_\_ /к.ф.-м.н. Токарева Н. А./

**Выпускная квалификационная работа допущена к защите** «\_\_» \_\_\_\_\_ 20\_\_ г.  
(дата)

**Заведующий кафедрой** \_\_\_\_\_ /д.т.н. Черемисина Е. Н./

*г. Дубна, 2017*

## Аннотация

Магистерская работа посвящена разработке среды освоения когнитивных облачных сервисов в виртуальной компьютерной лаборатории.

В ходе работы произведен обзор состояния когнитивных технологий, исследованы методы когнитивной аналитики, выделены классы задач, решаемых с помощью когнитивных технологий, сформированы технологические задачи для которых требуются знания когнитивных технологий, разработаны требования к системе, разработана архитектура системы освоения когнитивных технологий в среде ВКЛ. В результате работы были разработаны образы с предустановленным и настроенным программным обеспечением для выполнения учащимися практических заданий по созданию приложений, разработан демонстрационный набор программ на языке *C#*, использующих облачные когнитивные сервисы *Microsoft Cognitive Services* и *IBM Watson*, созданы видео-инструкции по использованию *API* когнитивных сервисов и обучающий веб-портал.

Автор работы: Логинов В.С. Работа выполнена под научным руководством доцента, к.т.н. Белова М.А. Работа состоит из введения, общей характеристики, трех глав, заключения и списка литературы. Работа изложена на 63 страницах, содержит 50 рисунков и 1 таблицу.

## Abstract

The master's work is devoted to the development of the environment for the development of cognitive cloud services in a virtual computer laboratory.

In the course of the work a review of the state of cognitive technologies, methods of cognitive analytics are explored, classes of problems solved with the help of cognitive technologies are identified, technological tasks for which knowledge of cognitive technologies are required, system requirements are developed, , the architecture of the cognitive technology development system in the virtual computer laboratory. As a result of work were created images with pre-installed and customized software for students to perform practical tasks on creating applications, a demonstration set of *C #* programs using *Microsoft Cognitive Services* and *IBM Watson*, created video instructions for the use of the *API* cognitive services and a training web portal.

Author: Loginov V.S. The work performed under the scientific supervision of. Belov M.A. The work consists of an introduction, a general characteristic, three chapters, a conclusion and a list of references. The work is set out on page 63, contains 50 drawings and 1 table.

# ОГЛАВЛЕНИЕ

<b>ВВЕДЕНИЕ .....</b>	<b>5</b>
<b>ОБЩАЯ ХАРАКТЕРИСТИКА РАБОТЫ .....</b>	<b>6</b>
<b>ГЛАВА 1. АНАЛИЗ ТЕКУЩЕГО СОСТОЯНИЯ ВИРТУАЛЬНОЙ КОМПЬЮТЕРНОЙ ЛАБОРАТОРИИ .....</b>	<b>8</b>
1.1.    Виртуальная компьютерная лаборатория .....	8
1.2.    Обоснование потребности совершенствования ВКЛ на основе анализа тенденций развития современных ИТ технологий. ....	10
1.3.    Текущие учебные задачи по формированию знаний, умений, владений учащихся .....	12
1.4.    Выбор облачных когнитивных сервисов .....	13
1.5.    Преимущества внедрения когнитивных сервисов в современное ПО .....	15
1.6.    Требования к освоению когнитивных сервисов .....	16
<b>ГЛАВА 2. АНАЛИЗ ТЕКУЩЕГО СОСТОЯНИЯ КОГНИТИВНЫХ ТЕХНОЛОГИЙ И КОГНИТИВНЫХ ОБЛАЧНЫХ СЕРВИСОВ .....</b>	<b>18</b>
2.1.    Анализ текущего состояния когнитивных технологий .....	18
2.2.    IBM Watson .....	20
2.3.    Microsoft Cognitive Services .....	22
2.4.    Сравнительный анализ облачных сервисов Microsoft Cognitive Services и IBM Watson ...	24
2.5.    Дидактические особенности освоения когнитивных технологий .....	28
2.6.    Сравнительный анализ дидактических методов обучения .....	30
2.7.    Общая архитектура системы освоения когнитивных технологий в среде ВКЛ .....	32
<b>ГЛАВА 3. РЕАЛИЗАЦИЯ СРЕДЫ ОСВОЕНИЯ КОГНИТИВНЫХ СЕРВИСОВ В ВИРТУАЛЬНОЙ КОМПЬЮТЕРНОЙ ЛАБОРАТОРИИ .....</b>	<b>35</b>
3.1.    Развертывание виртуальной машины в ВКЛ .....	35
3.2.    Развертывание программно-инструментальных средств .....	39
3.3.    Разработка демонстрационного набора программ .....	40
3.3.1. <i>Microsoft Cognitive Services</i> .....	40
3.3.2. <i>IBM Watson API</i> .....	51
3.4.    Создание видео-инструкций по использованию API когнитивных облачных сервисов .....	58
3.5.    Создание обучающего веб-портала .....	59
<b>ЗАКЛЮЧЕНИЕ .....</b>	<b>61</b>
<b>СПИСОК ЛИТЕРАТУРЫ .....</b>	<b>62</b>

## ВВЕДЕНИЕ

Из-за быстрого роста научно-технического прогресса возникает объективная необходимость совершенствования учебного процесса, повышения эффективности и качества подготовки профессиональных кадров. Обучение с использованием современных программных и аппаратных средств выводят учебный процесс на качественно новый уровень.

Аппаратные требования современных корпоративных информационных систем (ИС), требующие для функционирования мощные персональные компьютеры (ПК) – это наиболее распространенная преграда, возникающая для студента в процессе обучения. Поэтому людям, заинтересованным в получении новых знаний и опыта в различных ИТ-областях, часто оказывается проблематичным установить необходимое программное обеспечение на своем ПК. Программно-технологическая платформа виртуальной компьютерной лаборатории (ВКЛ) университета «Дубна» предоставляет доступ к вычислительным ресурсам и необходимому ПО независимо от местоположения пользователя. Применение ВКЛ помогает обучающимся лучше усваивать теоритический материал и предоставляет новые возможности для приобретения практических навыков.

Существенной особенностью информационной области является то, что в ней постоянно появляется множество новых технологий, аппаратных и программных продуктов, что значительно усложняет для вуза выбор из них наиболее значимых и существенных для конкретного направления подготовки студентов.

Одним из приоритетных направлений ИТ-технологий на сегодняшний день являются когнитивные технологии. Это не просто сфера исследований, а целая промышленность. Для таких мировых гигантов, как *IBM* и *Microsoft* облачные технологии и когнитивные вычисления стали приоритетными направлениями деятельности. Целью когнитивных вычислений является создание автоматизированных информационных систем, способных предвидеть новые проблемы и моделировать набор возможных решений, без участия человека.

Будущие ИТ-специалисты должны постоянно совершенствовать свои навыки и быть способными эффективно работать в изменившихся условиях глобального рынка, иначе они окажутся не востребованы на рынке труда после окончания университета. Поэтому актуальность темы научно-исследовательской работы не вызывает сомнений.

## ОБЩАЯ ХАРАКТЕРИСТИКА РАБОТЫ

*Целью магистерской диссертации* является разработка среды освоения когнитивных облачных сервисов в виртуальной компьютерной лаборатории.

Для достижения поставленной цели необходимо решить следующие *задачи*:

- Обзор состояния когнитивных технологий;
- Исследование методов когнитивной аналитики;
- Выделение классов задач, решаемых с помощью когнитивных технологий;
- Формирование технологических задач для которых требуются знания когнитивных технологий;
- Разработка требований к системе;
- Разработка архитектуры системы освоения когнитивных технологий в среде ВКЛ;
- Разработка образов с предустановленным и настроенным программным обеспечением для выполнения учащимися практических заданий по созданию приложений;
- Разработка демонстрационного набора программ на языке *C#*, использующих облачные когнитивные сервисы *Microsoft Cognitive Services* и *IBM Watson*;
- Создание видео-инструкций по использованию *API* когнитивных сервисов;
- Создание обучающего веб-портала.

*Объектом исследования* являются когнитивные облачные сервисы *IBM Watson Cognitive Services* и *Microsoft Cognitive Services*.

*Предмет исследования* – совершенствование технологии подготовки ИТ-специалистов с применением ВКЛ.

*Научная составляющая исследования* – анализ возможностей современных когнитивных систем и совершенствование виртуальной компьютерной лаборатории за счет внедрения среды освоения когнитивных облачных сервисов. Создание методических рекомендаций и технологии изучения разработки ПО с применением когнитивных облачных служб в среде ВКЛ.

*Практическая значимость исследования* заключается в том, что создание среды освоения когнитивных сервисов позволяет учащимся осваивать современные когнитивные сервисы в среде ВКЛ, а так же упростить процесс самоподготовки учащихся, за счет предоставления преднастроенной инфраструктуры и методического обеспечения, необходимых для получения практического опыта работы с когнитивными облачными сервисами.

***Апробация результатов исследования.*** Созданный обучающий веб-портал с наборами текстовых и видео инструкций по разработке приложений, использующих когнитивные облачные сервисы, был запущен и доступен для любого желающего. Студентам Института Системного Анализа и Управления было предложено создать свое приложение на основе представленных уроков. В качестве обратной связи была использована форма, которая отправляет отзывы на почту автору исследования. Результаты работы были одобрены, о чем свидетельствуют более чем 93% положительных отзывов. Но для более детального анализа требуется более продолжительный промежуток времени.

# **ГЛАВА 1. АНАЛИЗ ТЕКУЩЕГО СОСТОЯНИЯ ВИРТУАЛЬНОЙ КОМПЬЮТЕРНОЙ ЛАБОРАТОРИИ**

Данная работа развивает научные исследования Антипова О.Е., Белова М.А., Токаревой Н.А., Черемисиной Е.Н. в области внедрения ВКЛ в учебный процесс университета «Дубна».

## ***1.1. Виртуальная компьютерная лаборатория***

Виртуальная компьютерная лаборатория (ВКЛ) – это многокомпонентный комплекс программно-аппаратных средств, основанный на технологиях виртуализации, позволяющих предоставить вычислительные ресурсы университета для доступа к любым программным приложениям, используемым в учебном процессе университета, без установки и настройки на клиентском устройстве. В ВКЛ доступ к приложениям осуществляется посредством браузера или специального приложения, сети интернет и клиентских устройств (ПК, планшеты или мобильные устройства).

Компонентами архитектуры ВКЛ являются:

- Интегрированная система управления виртуальной компьютерной лабораторией, объединяющая в себе различные сервисы и подсистемы.
- Сервер повышенной надежности – центр обработки данных (ЦОД) лезвийной архитектуры с возможностью резервирования компонентов.
- Коммуникационные сетевые магистрали.
- Система резервного питания.
- Система резервного копирования.
- Система балансировки нагрузки.
- Гипервизоры.
- Виртуальные машины.
- Виртуальный компьютерный класс.
- Программное обеспечение.

Архитектура ВКЛ представлена на рис. ниже.

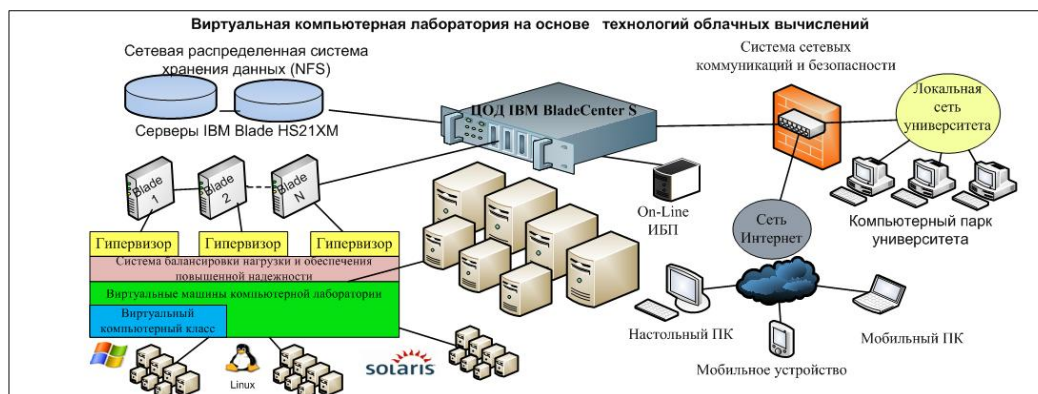


Рисунок 1. Архитектура виртуальной компьютерной лаборатории

Интегрированная система управления обеспечивает взаимосвязь перечисленных компонентов, предоставляя единый, централизованный интерфейс для доступа ко всем функциональным частям и компонентам ВКЛ через браузер. Эта система определяет возможности централизованного управления, мониторинга и удаленной работы. За счет семантической разметки страниц она позволяет с какой-либо открытой веб-страницы перейти на связанные с ней страницы, содержащие дополнительные или родственные сведения, а также настройки, необходимые для удобной, продуктивной и интуитивно-понятной работы в системе. Интегрированная система управления ВКЛ предоставляет удаленный доступ ко всем подсистемам, в том числе и аппаратным, позволяя администраторам лаборатории удаленно обновлять микропрограммы всех ее компонентов. Интегрированная система управления позволяет так же удаленно проводить диагностику аппаратных частей ВКЛ, перезагружать их или даже выключать, что обеспечивает мобильность в управлении и обслуживании всей аппаратной платформы виртуальной компьютерной лаборатории. [1]

Центральный доступ к программной платформе позволяет загружать и редактировать дистрибутивы программного обеспечения, операционных систем, интерактивных учебников, проводить контроль качества знаний, организовывать вебинары, контролировать усвояемость и успеваемость, формировать индивидуальные траектории обучения. [1]

Виртуальная компьютерная лаборатория, на сегодняшний день, стала неотъемлемой частью учебного процесса университета «Дубна».



## **1.2. Обоснование потребности совершенствования ВКЛ на основе анализа тенденций развития современных ИТ технологий.**

Информационные технологии не стоят на месте. Эксперты создают инновации, способные облегчить жизнь человека. Многие страны увидели возможности ИТ технологий приносить в экономику государства высокие дивиденды, что привело к повышению спроса на информационные услуги по всем миру.

Сейчас мировой ИТ-рынок не перестает быть самым динамичным в мире и постоянно развивается. Он играет высокую роль для современного общества и его влияние на человека только усиливается. [2]

Одним из самых перспективных направлений на мировом рынке информационных технологий сегодня считаются облачные технологии. Они представляют собой технологию, которая не требует использования локального сервера или компьютера и выполняется удаленно, например через интернет. Облачные технологии состоят из большого количества серверов, расположенных в ЦОД (центрах обработки данных). «Облака» обеспечивают ресурсами и неограниченными площадками для размещения данных огромного количества программных продуктов по всему миру. Особую популярность данная технология завоевала у предприятий и организаций. Предприятия по всему миру активно используют «облака» в целях экономии ресурсов, технология позволяет им экономить на размещении собственных *ERP*-систем (система планирования ресурсов предприятия) или *CRM*-систем (система управления взаимоотношениями с клиентами), содержание которых требует значительных затрат. Существенные усилия рынка РФ направлены на формирование отечественных решений в области «облачных вычислений». Многочисленные российские компании вкладывают ресурсы в освоение этой области. Сейчас создаются условия, способные обеспечить гибкий переход бизнеса в облака: развитие технологий виртуализации, конвергенции, опыт в создании объектных хранилищ и облачных порталов нивелируют те недостатки облаков, которые не устраивали ИТ-руководителей. Теперь использование облачных сервисов становится синонимом конкурентоспособности и главным средством решения задач мониторинга, эффективного обслуживания рабочих нагрузок, управления безопасностью и емкостью.

Виртуальная компьютерная лаборатория это облачный сервис в рамках университета. Она позволяет существенно снизить издержки на модернизацию и обслуживание компьютерного парка, а так же предоставляет студентам удаленный доступ, посредством локальной сети и сети интернет, к программно-технологическим платформам университета для решения всевозможных научно-исследовательских и вычислительных задач в любом месте и в любое время.

Опыт эксплуатации виртуальной компьютерной лаборатории на основе технологии облачных вычислений и виртуального компьютерного класса в учебном процессе университета «Дубна» в течение двух лет, а также анализ эффективности, проведенный, в том числе, и на основании опроса выпусков 2009–2010 гг. показал, что знания, умения и навыки, полученные в рамках инновационного учебного процесса, удовлетворяют всем требованиям ведущих компаний отрасли *ИТ*. Учащиеся смогли более эффективно проходить не только устные собеседования, но и демонстрировать свою высокую квалификацию на практике, решая реальные задачи, принимая участие в реальных проектах. За счет увеличения потенциала выпускников повысилась их востребованность на рынке труда, что позволило увеличить количество трудоустроенных выпускников в ведущие мировые компании. [3]

Подводя итоги важно отметить, что при использовании программно-технологической платформы ВКЛ и аналогичных разработок в современных университетах, основными принципами современного *ИТ*-образования становятся:

- «Образование через пространство и время» – концепция удаленного, индивидуального обучения, доступного в любое время и в где угодно;
- «Свобода действий и свобода выбора» – внедрение в учебный процесс различных свободно распространяемых программных продуктов и систем, интеграция новых платформ и инновационных технологий;
- Установка к поиску новых знаний, умений и навыков для достижения поставленной цели;
- Развитие навыков работы в командах специалистов различных областей;
- Постоянное творческое самосовершенствование.

Все эти концепции соответствуют изменяющимся условиям быстро развивающейся глобализации образования, развития систем дистанционного обучения, появления новых программно-технологических и аппаратных платформ и массового использования ресурсов сети интернет в целях образования. [3]

Одним из приоритетных направлений ИТ-технологий, на сегодняшний день, являются когнитивные вычисления. Это не просто сфера исследований, а целая промышленность. Для таких мировых гигантов, как *IBM* и *Microsoft* облачные технологии и когнитивные вычисления стали приоритетными направлениями деятельности.

Виртуальная компьютерная лаборатория должна постоянно совершенствоваться, чтобы помогать студентам соответствовать постоянно возрастающим требованиям к ИТ специалистам и быть востребованными на рынке труда после окончания университета.

### ***1.3. Текущие учебные задачи по формированию знаний, умений, владений учащихся***

В настоящее время конкурентоспособность человека на рынке труда во многом зависит от его способности овладеть новыми технологиями. Поэтому ведущая идея современного образования выражена в попытке увязать результирующую составляющую образования с планируемыми результатами развития ученика. Так возникла идея компетентностного подхода.

Компетентностный подход усиливает практическую ориентированность образования, подчеркивает роль опыта, умения на практике реализовать знания. Таким образом, в компетентностном подходе значение слова «знание» означает владение тем или иным комплексом предметных (специальных) и общих (ключевых) компетенций, позволяющих разрешать проблемные ситуации, при необходимости самостоятельно учиться и достигать общественных или личных целей.

Высококвалифицированный программист в области когнитивных технологий в логике компетентного подхода:

- Умеет выбрать нужную технологию для достижения желаемого результата;
- Работает над совершенствованием своих профессиональных навыков;
- Вписывается в стандарт современного профессионального программиста;
- Использует облачные когнитивные сервисы для улучшения своих приложений.

Для того, чтобы сформировать у программиста ЗУВ (знание, умение, владение) в области когнитивных технологий, необходимо создать инфраструктуру для обучения. ЗУВ могут быть использованы для создания приложений, решающих следующие задачи:

- Взаимодействие с людьми с использованием естественного языка;
- Понимание изображений и видеопотока;
- Идентификация пользователей с помощью фото или голоса;
- Анализ текста для выявления настроения клиентов и их намерений;
- Перевод текста и речи для создания приложений с поддержкой нескольких языков;
- Реализация системы рекомендаций на основе совершенных действий пользователя.

#### 1.4. Выбор облачных когнитивных сервисов

При выборе провайдера облачных когнитивных сервисов необходимо учитывать принцип неантогонистичности выстроенному учебному процессу в институте Системного Анализа и Управления. Необходимо опираться на уже имеющиеся у учащихся знания и компетенции. С начальных курсов студенты института САУ изучают языки программирования *C#* и *JAVA*, как наиболее популярные и перспективные для разработки корпоративного ПО, поэтому при выборе когнитивных сервисов, должна учитываться поддержка указанных языков программирования.

Рынок облачных когнитивных сервисов представлен такими мировыми гигантами как *IBM*, *Microsoft*, *Google*, *Amazon*.

Например *Amazon* предоставляет набор из 4 служб:

- *Amazon Lex* – создание текстовых и голосовых ботов для чатов;
- *Amazon Polly* – преобразование текста в живую речь;
- *Amazon Recognition* – поиск и анализ изображений;
- *Amazon Machine Learning* – использование технологий машинного обучения.

Доступ к ним осуществляется через готовые комплекты средств разработки *software development kit (SDK)*. Это удобно, если существует *SDK* для вашей платформы. Но компании *Google*, *IBM* и *Microsoft* пошли другим путем и представили более современное и универсальное решение, это использование *REST API* для доступа к своим облачным когнитивным службам.

Использование *REST API*, по сравнению с *SDK*, для разработчиков имеет ряд явных преимуществ. Во-первых, доступ к службе программируется только один раз, нет необходимости писать код для каждой платформы. Во-вторых, документация пишется лишь для одного сервиса, а не для каждого *SDK*. В-третьих, если необходимо внести изменения в код или документацию, то сделать это нужно только в одном месте. И самым главным достоинством *REST* является то, что работать с ним может любая платформа.

*Google*, на момент написания этой научно-исследовательской работы, имеет набор из 6 служб, которые представлены на рис. ниже.

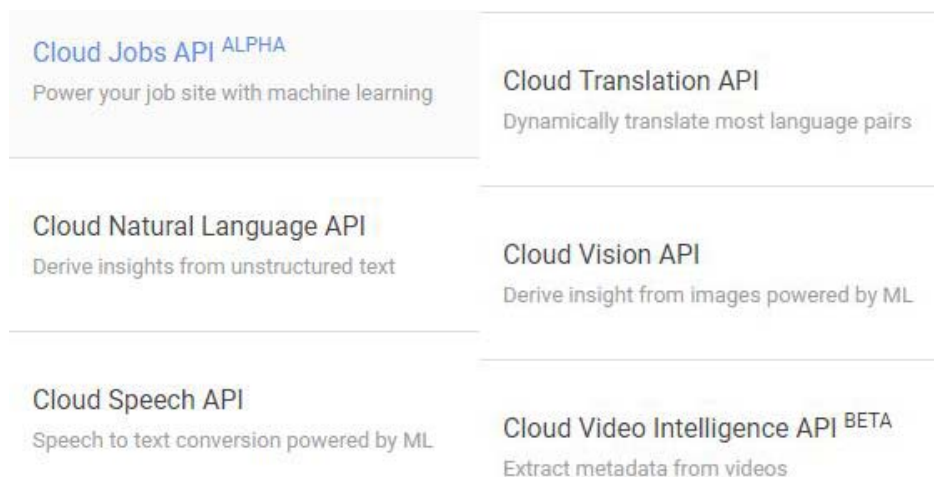


Рисунок 2. Когнитивные облачные службы *Google Cloud Platform*

Самый широкий набор сервисов предоставляют *Microsoft Cognitive Services* (более 30) и *IBM Watson* (более 15). Кроме того, к продуктам *IBM Watson* есть бесплатный доступ в рамках партнерского соглашения и центра компетенции *IBM* в структуре Института Системного Анализа и Управления, а продукты Майкрософт широко используются в учебном процессе университета «Дубна». Именно поэтому для исследования были выбраны два облачных когнитивных сервиса:

- *IBM Watson Services*;
- *Microsoft Cognitive Services*.

Для написания приложений, использующих службы этих сервисов, была выбрана платформа *.Net* и язык *C#*. Он схож с *Java*, оба являются строго типизированными, объектными. Они вобрали многое из синтаксиса *C++*, но в отличие от *C++* проще в освоении. Научившись использовать облачные когнитивные сервисы в своих приложениях на языке *C#*, студентам будет проще освоить тоже самое на *Java*. Эти знания позволят будущим *IT*-специалистам разрабатывать «умные» мобильные приложения.

### **1.5. Преимущества внедрения когнитивных сервисов в современное ПО**

Сфера применения когнитивных систем обширна. Уже сейчас они используются в медицине, экономике и даже кулинарии.

В бизнесе когнитивные технологии позволяют обнаруживать проблемные места в инфраструктуре, расписании дня предприятия и прочих элементах. Благодаря ликвидации этих проблем экономятся значительные средства и время сотрудников. Кроме того, понимая аспекты каждого конкретного человека, когнитивная система может осуществлять индивидуальное взаимодействие с покупателями. На основе транзакций клиента, она может порекомендовать товар, который нужен покупателю. Банки используют методы идентификации мошенничества с банковскими картами, системы распознавания речи для автоматизации работы службы поддержки клиентов и для идентификации клиентов. Ритейлеры применяют эти технологии для маркетинга и проведения промо-акций.

В здравоохранении эти системы, прорабатывая огромное количество данных позволяют поставить тот или иной диагноз с определенной вероятностью. Особенно это актуально в сложных случаях, например, при раковых заболеваниях.

В кулинарии когнитивные системы могут составлять рецепты различных блюд, исходя из заранее заданного списка продуктов.

В спорте когнитивные вычисления помогают в режиме реального времени проводить оценку тренировки различных спортсменов, а также набирать команды игроков с необходимыми тренеру характеристиками. И это все лишь малая часть того, на что способны когнитивные системы.

По словам представителей *IBM*, уже через десять лет не останется такой индустрии, которую обойдут стороной когнитивные технологии. Согласно отчету аналитической фирмы *International Data Corporation*, к 2018 году половина всех пользователей будет взаимодействовать с когнитивными технологиями на регулярной основе. [8]

Когнитивные вычисления это тренд последних лет, который помогает человеку справляться с огромным потоком неструктурированной информации, поэтому нет сомнений, что их нужно изучать.

## 1.6. Требования к освоению когнитивных сервисов

Для освоения когнитивных облачных сервисов необходимы вычислительные ресурсы и определенный набор программ. Для того, чтобы студенты могли комфортно осваивать эти сервисы, были выделены ключевые требования:

- Функциональные:
  - Система должна предоставлять доступ к учебным материалам из любого места, где есть интернет и в удобное для студента время.
  - Среда освоения должна содержать текстовые и видео-инструкции по разработке «умных» приложений;
  - Система должна предоставлять возможность студенту индивидуально ознакомиться с материалом в максимально удобной форме, повторно пройти весь курс или какой-то проблемный фрагмент;
  - Среда освоения должна предоставлять возможность посмотреть готовые решения, если возникли какие-то трудности при разработке;
  - Среда освоения должна обеспечить студента всем необходимым для изучения когнитивных облачных сервисов.
- Операционные:
  - Необходима среда разработки для создания приложений на языке C#;
  - Необходима среда разработки для создания мобильных приложений на языке программирования *Java*;
  - Необходим эмулятор для тестирования мобильных приложений под *Android*.
- Лингвистические:
  - Используемое при разработке ПО должно иметь широкое распространение и использоваться в промышленных масштабах.
  - Машина-тренажер должна быть высокопроизводительной для комфортной работы, операционная система должна быть не старше *Windows 7*, объем оперативной памяти 4 Гб и более, ЦП – 2 ГГц (x64) и выше, место на жестком диске 128 Гб и более.

После создания среды освоения сервисов, удовлетворяющих этим требованиям, студент сможет:

- Посмотреть готовый шаблон программы, использующей когнитивные сервисы;

- Изучить службы сервиса, используя текстовые и видео инструкции;
- Использовать машину тренажер для создания приложений, использующих облачные когнитивные сервисы;
- Тестировать свое приложение, используя машину-тренажер.

*Use case* диаграмма соответствующих сценариев представлена на рис. ниже.

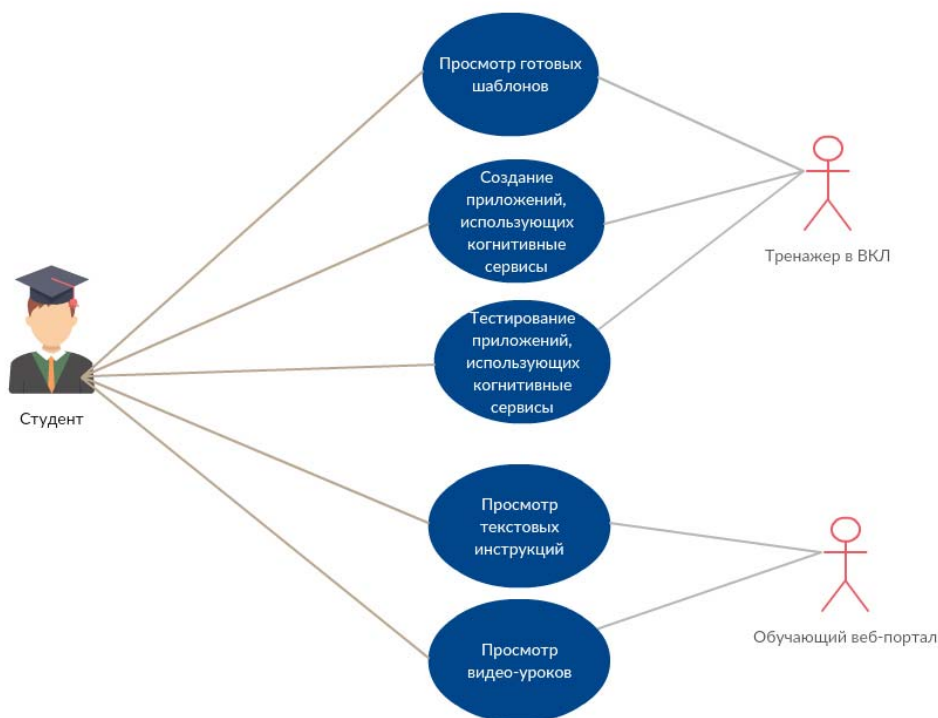


Рисунок 3. *Use case* диаграмма

На основании проведенного анализа можно сделать вывод, что для повышения качества образования студентов *ИТ*-сферы необходимо использовать самые современные инструменты обучения. Успешный опыт применения программно-технологической платформы ВКЛ в университете «Дубна» доказал, что она является одним из таких инструментов. Но прогресс не стоит на месте, и поэтому виртуальная компьютерная лаборатория должна постоянно совершенствоваться, чтобы помогать студентам соответствовать возрастающим требованиям к *ИТ*-специалистам и быть востребованными на рынке труда после окончания университета.

Одним из приоритетных *ИТ*-направления на сегодняшний день являются когнитивные технологии. Они помогают человеку справляться с огромным потоком неструктурированной информации, поэтому нет сомнений, что их нужно изучать. Внедрение среды освоения когнитивных облачных сервисов в знаниевое пространство ВКЛ, способствует совершенствованию программно-технологической платформы и самое главное – повышению качества образования студентов университета «Дубна».



## **ГЛАВА 2. АНАЛИЗ ТЕКУЩЕГО СОСТОЯНИЯ КОГНИТИВНЫХ ТЕХНОЛОГИЙ И КОГНИТИВНЫХ ОБЛАЧНЫХ СЕРВИСОВ**

Главным предназначением когнитивных технологий является упрощение работы человека со своим информационным окружением.

### ***2.1. Анализ текущего состояния когнитивных технологий***

Со времен зарождения компьютеров задача инженеров, проектировщиков и разработчиков состояла том, чтобы наделить машины интеллектом и способностью мыслить, как человек. Сегодня огромный рост вычислительных мощностей и значительный прогресс в области искусственного интеллекта делают эту задачу более реальной. Растущая сложность вычислений, огромные объемы неструктурированных данных, а также необходимость в более интеллектуальных методах автоматизации открывают путь когнитивным системам.

Когнитивные вычисления – это тренд последних лет, который развивается быстрыми темпами и помогает человеку справляться с огромным потоком информации. Каждый день человечество генерирует около 2,5 квинтиллионов байтов данных, и 80% из них являются неструктурированными, а значит невидимыми для обычных ПК. На помощь приходят когнитивные вычисления, технологии, которые частично повторяют особенности работы мозга человека и способные работать во много раз эффективнее. Они способны упростить жизнь как отдельно взятому человеку, так и бизнес-структурам.

Уже сейчас, используются такие системы в огромном множестве различных сфер и направлений, включая банки, материаловедение, оптимизацию бизнеса, управление инфраструктурой городов, оценку состояния окружающей среды, исследования в различных областях науки и в медицине. Основная задача когнитивных технологий — дать человеку возможность работать с неструктурированными данными удобным ему способом. При этом постепенно создаются системы нового типа, которые не просто следуют заданному алгоритму, а способны учитывать множество сторонних факторов во время работы, самообучаться, использовать результаты прошлых вычислений и сторонние ресурсы (например интернет).

Для того, чтобы соответствовать своему предназначению (упрощение работы человека со своим информационным окружением), когнитивные системы должны быть:

- Адаптивными (должны уметь обрабатывать динамические данные и предоставлять результат в режиме реального времени или близко к этому);

- Интерактивными (должны взаимодействовать с пользователем таким образом, чтобы он чувствовал себя комфортно, получая нужный результат);
- Самообучаемыми (должны основываться не только на новых данных, но и «запоминать» предыдущие итерации, и обращаться к этой информации при необходимости);
- Контекстуальными (должны понимать, идентифицировать и выделять контекстуальные элементы, такие как значение, время, местоположение, профиль пользователя, цель, процесс и задачу).

Особое внимание стоит уделить когнитивным сервисам. Это набор алгоритмов машинного обучения, которые разрабатываются мировыми гигантами такими как *Microsoft*, *IBM*, *Google* и другими для решения проблем в области искусственного интеллекта. Целью когнитивных сервисов является демократизация искусственного интеллекта путем его упаковки в дискретные компоненты. Облачные сервисы представляют собой набор готовых *REST API*, в доступной форме дающих разработчикам всю мощь алгоритмов машинного зрения, анализа естественного языка и распознавания голоса для использования в своих приложениях.

Стоит отметить, что доступность сервисов в виде *REST API* позволяет использовать его на совершенно любых платформах и с помощью своих любимых технологий разработки. *API* сервисов довольно прост и не требует знаний в области *data science*. Все трудоемкие вычисления происходят на стороне сервиса, поэтому требования к производительности клиентского устройства минимальны.

В рамках научно-исследовательской работы был произведен обзор двух крупных когнитивных систем.

## 2.2. *IBM Watson*

На основании анализа литературных источников [12, 13, 14] становится очевидно, что одной из первых когнитивных систем в мире, которая способна понимать, делать выводы и обучаться, стал суперкомпьютер *IBM Watson*. Он позволяет преобразовывать целые отрасли, различные направления науки и техники. Например, предсказывать появление эпидемий или возникновения очагов природных катастроф в различных регионах, вести мониторинг состояния атмосферы больших городов, оптимизировать бизнес-процессы, узнавать, какие товары будут в тренде в ближайшее время.

Сейчас на основе этой облачной *API*-платформы разрабатываются различные приложения, сервисы и процессы.

Ниже представлен неполный список задач, которые можно решать с помощью сервиса *IBM Watson*:

1. ***Vision APIs*** – распознавание того, что находится на фото или видеокадре.
  - 1.1. *Visual Recognition*: распознавание образов по изображениям. Поиск лиц, определение возраста, пола и поиск похожих лиц в коллекции.
2. ***Speech APIs*** – распознавание голоса (*speech to text*) и синтез речи (*text to speech*).
  - 2.1. *Speech to Text*: распознавание голоса и перевод в текст.
  - 2.2. *Text to Speech*: перевод текста в речь.
3. ***Language APIs*** – определение тональности текста, выделение ключевых фраз, обработка голосовых команд, перевод текста на другой язык, общение с клиентами на естественном языке (боты).
  - 3.1. *Alchemy Language*: обработка естественного языка, понимание настроения, ключевых слов и т.д.
  - 3.2. *Conversation*: позволяет быстро создавать, тестировать и разворачивать ботов.
  - 3.3. *Dialog*: позволяет разработчику автоматизировать диалог между пользователем и приложением на естественном языке.
  - 3.4. *Document Conversion*: преобразование документов в другие форматы.
  - 3.5. *Language Translator*: переводчик текста с одного языка на другой.
  - 3.6. *Natural Language Classifier*: интерпретирование намерений автора текста.
  - 3.7. *Natural Language Understanding*: извлечение из текста ключевых слов, чувств, эмоций и т.д.

- 3.8. *Personality Insights*: извлечение характеристик личности, основываясь на том, как пишет человек.
- 3.9. *Retrieve and Rank*: поиск наиболее актуальной информации из коллекции документов.
- 3.10. *Tone Analyzer*: использует лингвистический анализ для обнаружения эмоций, социальных тенденций и стиля написания.
4. ***Data Insights APIs*** – позволяет принимать решения, рассматривая несколько целей.
- 4.1. *Alchemy Data News*: поиск новостей за последние  $N$  дней.
- 4.2. *Discovery*: когнитивный поиск и анализ контента.
- 4.3. *Discovery News*: «умный» поиск новостей.
- 4.4. *Tradeoff Analytics*: помогает сделать лучший выбор на основе предоставленных данных.

## 2.3. *Microsoft Cognitive Services*

Основным конкурентом *IBM* на рынке когнитивных систем является компания *Microsoft*, которая разработала продукт *Microsoft Cognitive Services*. Использование *API* этих служб позволяют создавать приложения, которые могут слышать, видеть, говорить и понимать с помощью естественных методов коммуникации, используя всего несколько строк кода. Они работают на разных платформах и постоянно совершенствуются.

*API Microsoft Cognitive Service* сгруппированы в пять категорий:

1. ***Vision APIs*** – анализ изображений и видео контента для идентификации лиц и эмоций, а так же для обнаружения информации для принятия решений. Включает в себя:
  - 1.1. *Face*: идентификация людей на изображении, выделение лиц, определение возраста и пола.
  - 1.2. *Computer Vision*: определение объектов, представленных на изображении.
  - 1.3. *Emotion*: определение эмоций на изображении.
  - 1.4. *Video*: стабилизация видеопотока, определение движения, отслеживание лиц.
  - 1.5. *Content Moderator*: машинная модерация текста и изображений, обнаружение потенциально оскорбительного и нежелательного контента.
2. ***Speech APIs*** – преобразование текста в речь, распознавание естественной речи, идентификация говорящего. Включает в себя:
  - 2.1. *Bing Speech*: преобразование речи в текст, текста в речь.
  - 2.2. *Custom Speech Service*: создание моделей речи для транскрипции речи в текст.
  - 2.3. *Speaker Recognition*: распознавание говорящего.
3. ***Language APIs*** – восприятие естественного языка, т.е. обнаружение и исправление орфографических ошибок, распознавание голосовых команд и анализ сложного текста, в том числе эмоциональной окраски и ключевых фраз. Включает в себя:
  - 3.1. *Bing Spell Check*: проверка правописания.
  - 3.2. *Language Understanding Intelligent Services (LUIS)*: интеллектуальная служба распознавания речи – позволяет общаться с приложением на естественном языке.
  - 3.3. *Linguistic Analysis*: синтаксический анализ языка.

- 3.4. *Text Analytics*: выделение ключевых фраз, определение настроения в неструктурированном тексте и многое другое.
- 3.5. *Web Language Model*: преобразование хеш-тегов, *url* в обычный текст.
4. **Knowledge APIs** – расширение знаний клиентов за счет нахождения персонализированных рекомендаций по товарам, мероприятиям, местам, а также академическим статьям и журналам. Включает в себя:
- 4.1. *Academic Knowledge*: использование знаний, накопленных в *Microsoft Academic Graph*.
- 4.2. *Entity Linking Intelligence Service*: поиск объектов для связывания в абзаце.
- 4.3. *Knowledge Exploration Service*: эффективный способ добавления интерактивного поиска (предложение авто завершений фраз).
- 4.4. *Recommendations*: рекомендация товаров или услуг, на основе прошлой деятельности клиента.
5. **Search APIs** – основаны на *Bing* и позволяют реализовать мощные средства поиска в приложениях. Включает в себя:
- 5.1. *Bing Autosuggest*: автозаполнение на основе части слова.
- 5.2. *Bing Image Search*: поиск изображений по запросу.
- 5.3. *Bing News Search*: поиск новостей по запросу.
- 5.4. *Bing Video Search*: поиск видео по запросу.
- 5.5. *Bing Web Search*: поиск изображений, новостей, видео и т.д. в одном запросе.

## 2.4. Сравнительный анализ облачных сервисов *Microsoft Cognitive Services* и *IBM Watson*

Для сравнения двух облачных когнитивных сервисов была выполнена классификация служб *API* по 11 типам. Результаты представлены в таблице 1.

Таблица 1. Соответствие *API Microsoft Cognitive Services* и *IBM Watson Services*

Тип	IBM Watson Cognitive Services	Microsoft Cognitive Services	Описание
Компьютерное зрение	<p><b>Visual Recognition:</b> распознавание образов на изображении, поиск лиц, определение приблизительного возраста и пола, а так же поиск похожих изображений в коллекции.</p> <p>Отличительной особенностью этой службы является то, что можно создавать индивидуальный классификатор изображений, обучая его на собственном материале.</p>	<p><b>Computer Vision:</b> анализ и описание того, что находится на изображении с помощью тегов.</p> <p><b>Face:</b> обнаружение и распознавание лиц на фотографиях, определение пола и оценка возраста, а также оценка схожести лиц, поиск похожих лиц на серии фотографий по образцу, распознавание людей на основе обучающей выборки.</p> <p><b>Emotion:</b> определение эмоций на фотографии.</p> <p><b>Content Moderator:</b> машинная модерация текста и изображений, обнаружение потенциально оскорбительного и нежелательного контента.</p>	<p><i>Microsoft Cognitive Services</i>, в отличие от <i>IBM Watson</i> разделяет визуальное распознавание на четыре различных <i>API</i>.</p>

Перевод текста	<p><b>Language Translator:</b> перевод текста с одного языка на другой. Сервис предлагает несколько предметно-ориентированных моделей, которые можно настроить на основе выбранной уникальной терминологии.</p>	<p><b>Translator Text:</b> перевод текста с поддержкой нескольких языков.</p> <p><b>Translator Speech:</b> устный перевод с поддержкой нескольких языков</p>	<p><b>Microsoft</b> предоставляет возможность перевода устной речи, но у <b>IBM</b> есть служба <b>Speech to Text API</b> для конвертации устной речи в текст, т.е. используя два <b>API</b> можно выполнить устный перевод.</p>
Анализ текста	<p><b>Alchemy Language:</b> извлечение из текста ключевых слов, связей и настроения текста.</p> <p><b>Tone Analyzer:</b> использование лингвистического анализа для обнаружения в тексте: эмоций, соц. тенденций и стиля письма.</p> <p><b>Natural Language Understanding:</b> анализ семантических особенностей текста, таких как категории, понятия, эмоции, ключевые слова, настроения и т.д.</p>	<p><b>Text Analytics:</b> определение тональности, ключевых слов, и языка текста.</p> <p><b>Linguistic Analysis:</b> лингвистический анализ текста.</p> <p><b>Bing Spell Check:</b> проверка орфографии текста.</p> <p><b>Web Language Model:</b> разбивка строки без пробелов (например хештеги или <b>URL</b>) на слова.</p>	



	<p><b>Personality Insights:</b> извлечение характеристик личности, на основе того, как пишет человек.</p>		
Создание ботов	<p><b>Retrieve and Rank:</b> поиск наиболее релевантной информации из сборника документов.</p> <p><b>Natural Language Classifier:</b> понимание намерения текста. Используется для ответов на вопросы в контактном центре, создания чат-ботов и т.д.</p> <p><b>Conversation:</b> API для создания чат-ботов, понимающих естественный язык.</p>	<p><b>QnA Maker:</b> позволяет быстро создавать, обучать и публиковать пары вопрос-ответ для создания ботов на основе FAQ, документов и т.д.</p> <p><b>Language Understanding Intelligent Service (LUIS):</b> создание собственных языковых моделей, благодаря которым любое приложение или бот может понимать команды пользователей.</p>	
Поиск информации	<p><b>Alchemy Data News:</b> поиск новостей за последние <math>N</math> дней.</p> <p><b>Discovery:</b> когнитивный поиск и анализ контента.</p> <p><b>Discovery News:</b> «умный» поиск новостей.</p>	<p><b>Academic Knowledge:</b> использование знаний, накопленных в <i>Microsoft Academic Graph</i>.</p> <p><b>Bing Image Search:</b> поиск изображений по запросу.</p> <p><b>Bing News Search:</b> поиск новостей по запросу.</p> <p><b>Bing Video Search:</b> поиск видео по запросу.</p> <p><b>Bing Web Search:</b> получение ссылок на веб-страницы по запросу.</p>	<p><i>Microsoft</i> фактически предоставляет доступ к поисковику <i>Bing</i> через свои API.</p>
Обработка документов	<p><b>Document Conversion:</b> преобразование</p>	Отсутствует	

	документов в другой формат.		
Конвертация речи в текст и обратно	<p><b><i>Speech to Text:</i></b> преобразование речи в текст.</p> <p><b><i>Text to Speech:</i></b> преобразование текста в речь.</p>	<p><b><i>Bing Speech:</i></b> преобразование речи в текст и обратно, а также выделение сути высказывания.</p>	
Обработка видео	Отсутствует	<p><b><i>Video:</i></b> интеллектуальная обработка видео, включает в себя: стабилизация «дрожащего» изображения в видео, выявление движения, обнаружение и отслеживание лиц, распознавание объектов в режиме реального времени.</p>	
Идентификация по голосу	Отсутствует	<p><b><i>Speaker Recognition:</i></b> идентификация говорящего.</p>	
Помощь в принятии решений	<p><b><i>Tradeoff Analytics:</i></b> помогает людям принять решение при выборе из нескольких целей.</p>	<p><b><i>Custom Decision Service:</i></b> пользовательская служба принятия решений рассматривает все возможные варианты и принимает решение.</p>	
Система рекомендаций	Отсутствует	<p><b><i>Recommendations:</i></b> рекомендация товаров, которые нужны клиентам на основе предыдущих транзакций.</p>	

## **2.5. Дидактические особенности освоения когнитивных технологий**

Дидактика – это раздел педагогики, раскрывающий закономерности усвоения знаний, умений, навыков и формирования убеждений. Существуют «частные дидактики» - методики обучения отдельным учебным предметам, отдельных категорий учащихся, в разных типах учебных заведений и формах образования. Предметом дидактики является связь преподавания и учения, их взаимодействие.

Задачи дидактики:

- Описание и объяснение процесса обучения и условий его реализации;
- Разработка более современных процессов обучения;
- Организация учебного процесса;
- Создание новых обучающих систем;
- Применение новых технологий обучения.

Так как в настоящее время процесс информатизации проявляется во всех сферах человеческой деятельности, использование современных информационных технологий является необходимым условием развития более эффективных подходов к обучению и совершенствованию методик преподавания. Их применение способствует повышению мотивации обучения учащихся, экономии времени, а интерактивность и наглядность способствует лучшему представлению, пониманию и усвоению учебного материала.

Основным средством ИКТ для информационной среды системы образования является персональный компьютер, возможности которого определяются его производительностью и установленным на нем программным обеспечением.

Существенной особенностью информационной области является то, что в ней постоянно появляется множество новых технологий, аппаратных и программных продуктов, что значительно усложняет для вуза выбор из них наиболее значимых и существенных для конкретного направления подготовки студентов. В результате существенной рассогласованности реальной потребности рынка в области ИТ-технологий и реального уровня подготовки выпускников, с одной стороны, предприятия ИТ-сферы испытывают «кадровый голод», а с другой, довольно большой процент выпускников по информационным направлениям подготовки после окончания вуза идут работать не по специальности. Основной тезис работодателей заключается в том, что для российского образования в целом характерен следующий дисбаланс: очень высокий уровень теоритических знаний при явной недостаточности практической составляющей. В итоге выделяются две ключевых проблемы в подготовке специалистов: недостаток практических навыков и излишняя «обобщенность» теоритических знаний (тогда как работодатели

хотели бы, чтобы специалист, прежде всего, владел глубокими, но узкоспециальными знаниями).

Когнитивные технологии неразрывно связаны с компьютерами, поэтому для их освоения необходим определенный набор программно-инструментальных и аппаратных средств. Но программное обеспечение часто бывает дорогостоящим и требует немалого времени для развертывания. Кроме того, для комфортного изучения когнитивных технологий нужны немалые аппаратные мощности. Но не у каждого студента есть возможность иметь производительный компьютер и тратить средства на лицензионное ПО.

Кроме того, для того, чтобы студенты могли в любой момент получить доступ к обучающим материалам, будет создан портал с инструкциями по разработке программ, использующих когнитивные облачные сервисы. Каждая инструкция будет подкреплена интерактивным видеоуроком, для того чтобы каждый ученик мог выбрать присущий ему темп обучения. В любой момент времени он сможет обратиться к той части информации, которая вызывает у него затруднения и посмотреть еще раз. Все уроки будут проиллюстрированы примерами, так как это повышает качество усвоения на 30-60%.

Применение модуля освоения когнитивных технологий в виртуальной компьютерной лаборатории университета «Дубна» включает в себя следующие дидактические особенности:

- Возможность получения доступа к учебным материалам из любого места, где есть интернет и в удобное для студента время;
- Детальное описание текстовых и видео инструкций по разработке программ;
- Возможность индивидуально ознакомиться с материалом в максимально удобной форме, повторно пройти весь курс или какой-то проблемный фрагмент;
- Минимальные требования к техническому обеспечению студентов;
- Экономия учебного времени (не нужно тратить время на установку программно-инструментальных средств).

## **2.6. Сравнительный анализ дидактических методов обучения**

Дидактические методы обучения – это способы совместной теоретической и практической деятельности преподавателей и студентов по достижению дидактических целей и задач. Можно условно выделить четыре группы методов:

1. Объяснительно-иллюстративный или репродуктивный;
2. Проблемное изложение (решение задач, вопросов, ситуаций);
3. Частично-поисковый, или эвристический (овладение отдельными этапами, элементами процесса научного поиска, познания);
4. Исследовательский (проблема решается учащимися самостоятельно, но под руководством учителя).

В репродуктивном методе обучения выделяются следующие признаки:

- Знания учащимся предлагаются в «готовом» виде;
- Учитель сообщает знания и объясняет их;
- Учащиеся сознательно усваивают знания, понимают их и запоминают. Критерием усвоения является правильное воспроизведение (репродукция) знаний;
- Необходимая прочность усвоения обеспечивается путем многократного повторения знаний.

Главное преимущество данного метода – экономность. Он обеспечивает возможность передачи значительного объема знаний, умений за минимально короткое время и с небольшими затратами усилий.

Человеческая деятельность может быть репродуктивной, исполнительской или творческой. Репродуктивная деятельность предшествует творческой, поэтому игнорировать ее в обучении нельзя, как нельзя и чрезмерно увлекаться ею. Репродуктивный метод должен сочетаться с другими методами.

Метод проблемного изложения является переходным от исполнительской к творческой деятельности. На определенном этапе обучения учащиеся еще не в силах самостоятельно решать проблемные задачи, а потому учитель показывает путь исследования проблемы, излагая ее решение от начала до конца. И хотя учащиеся при таком методе обучения не участники, а всего лишь наблюдатели хода размышлений, они получают хороший урок разрешения познавательных затруднений.

Сущность частично-поискового (эвристического) метода обучения выражается в следующих его характерных признаках:

- Знания учащимся не предлагаются в «готовом» виде, их нужно добывать самостоятельно;
- Учитель организует не сообщение или изложение знаний, а поиск новых знаний с помощью разнообразных средств;
- Учащиеся под руководством учителя самостоятельно рассуждают, решают возникающие познавательные задачи, создают и разрешают проблемные ситуации, анализируют, сравнивают, обобщают, делают выводы и т. д., в результате чего у них формируются осознанные прочные знания.

Метод получил название частично-поискового потому, что учащиеся не всегда могут самостоятельно решить сложную учебную проблему от начала и до конца. Поэтому учебная деятельность развивается по схеме: учитель - учащиеся - учитель - учащиеся и т. д. Одной из модификаций данного метода является эвристическая (открывающая) беседа.

Сущность исследовательского метода обучения сводится к тому, что:

- Учитель вместе с учащимися формулирует проблему, разрешению которой посвящается отрезок учебного времени;
- Знания учащимся не сообщаются. Учащиеся самостоятельно добывают их в процессе разрешения (исследования) проблемы, сравнения различных вариантов получаемых ответов. Средства для достижения результата также определяют сами учащиеся;
- Деятельность учителя сводится к оперативному управлению процессом решения проблемных задач;
- Учебный процесс характеризуется высокой интенсивностью, учение сопровождается повышенным интересом, полученные знания отличаются глубиной, прочностью, действенностью.

Исследовательский метод обучения предусматривает творческое усвоение знаний. Его недостатки – значительные затраты времени и энергии учителей и учащихся. Применение исследовательского метода требует высокого уровня педагогической квалификации.

В качестве базового метода освоения когнитивных облачных сервисов будет использоваться репродуктивный метод обучения, т.к. необходимо передать значительный объем знаний, умений за короткий промежуток времени и с минимальными затратами для студентов.

## 2.7. *Общая архитектура системы освоения когнитивных технологий в среде ВКЛ*

Образование не всегда идет в ногу со временем, выстраивая модели обучения в традиционной форме, и не обеспечивая тем самым, получение необходимых и актуальных знаний, умений и навыков. Исходя из этого, становится очевидно, что система высшего профессионального образования сегодня требует новых форм и методов преподавания.

IT-образование отличается непрерывным и быстрым ростом требований к квалификации специалистов. Основные принципы IT-образования в университете «Дубна» включают расширение направлений IT-подготовки и совершенствование программ учебных курсов и методов для интеграции современных информационных и образовательных технологий для повышения качества и доступности образования.

В качестве нового направления IT-подготовки в университете «Дубна» предлагается освоение когнитивных облачных сервисов. Чтобы научить программистов ЗУВ в области когнитивных технологий, необходимо создать инфраструктуру для обучения. Модель этой системы будет состоять из компонентов:

1. Набор текстовых и видео инструкций по созданию программ;
2. Демонстрационный набор программ на языке C#, использующих облачные когнитивные сервисы *Microsoft Cognitive Services* и *IBM Watson*;
3. Обучающий веб-портал;
4. Виртуальная машина-тренажер.

В архитектуре ВКЛ она будет выглядеть как на рис. ниже.



Рисунок 4. Архитектура ВКЛ вместе с модулем освоения когнитивных сервисов

Все эти ресурсы будут размещены в виртуальной компьютерной лаборатории и доступны удаленно. Очевидным преимуществом здесь является то, что студентам для того, чтобы приступить к изучению когнитивных облачных сервисов, достаточно иметь лишь ПК с доступом в сеть Интернет.

В виртуальной машине-тренажере будут установлены две среды разработки, которые позволят студентам разрабатывать не только *desktop*, но и мобильные приложения, использующие «умные» технологии.

На основании анализа облачных когнитивных технологий можно сделать вывод, что использование облачных сервисов *IBM Watson Services* и *Microsoft Cognitive Services* в своих приложениях позволит программистам решать следующие задачи:

- Распознавание текста и речи;
- Распознавание лиц, эмоций, объектов на изображениях и в видеопотоке;
- Идентификация человека по фото или голосу;
- Анализ текста;
- Перевод текста и речи;
- Помощь в принятии решений;
- Система рекомендаций.

Изучение этих сервисов полностью перекрывают требования ЗУВ, которые должны знать будущие ИТ-специалисты. Но т.к. когнитивные технологии неразрывно связаны с



компьютерами, для их освоения необходим определенный набор программно-инструментальных и аппаратных средств. Чтобы студент мог быстро приступить к изучению облачных когнитивных сервисов, не был зависим от программного обеспечения его ПК, а так же мог делать это в удобное для него время необходимо создать модуль освоения когнитивных технологий в виртуальной компьютерной лаборатории университета «Дубна». Внедрение этого модуля позволит студентам в удобной форме изучать службы этих сервисов, чтобы потом использовать их в своих разработках.

# ГЛАВА 3. РЕАЛИЗАЦИЯ СРЕДЫ ОСВОЕНИЯ КОГНИТИВНЫХ СЕРВИСОВ В ВИРУТАЛЬНОЙ КОМПЬЮТЕРНОЙ ЛАБОРАТОРИИ

## 3.1. Развертывание виртуальной машины в ВКЛ

Для доступа к вычислительным ресурсам университета необходим «*VMware vSphere Client*». При запуске клиента нужно указать *IP* адрес сервера, а так же реквизиты пользователя. (Рис. 5)



Рисунок 5. Окно авторизации *VMware vSphere Client*

Для создания новой виртуальной машины необходимо выбрать *File-New-Virtual Machine* или нажать *Ctrl + N* на клавиатуре для запуска мастера настройки, как на рис. ниже.

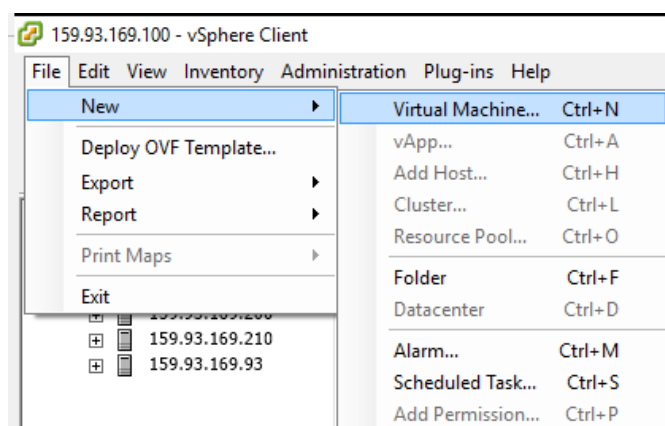


Рисунок 6. Создание новой виртуальной машины

На первой странице мастера настройки выбираем пользовательскую установку. После выбора способа установки необходимо настроить следующие параметры:

- Хранилище данных (место, где будет находиться *ВМ*);

- Операционная система;
- Количество используемых виртуальной машиной процессоров;
- Объем оперативной памяти;
- Назначение сетевой карты;
- Размер виртуальных дисков.

Первый шаг заключается в выборе кластера, где будет находиться *ВМ*. (Рис. 7)

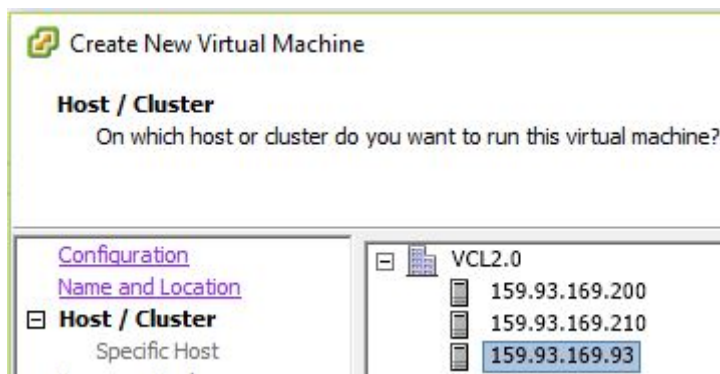


Рисунок 7. Выбор кластера для виртуальной машины

После этого необходимо выбрать хранилище данных, где будет размещаться виртуальная машина. Процесс выбора представлен на рис. ниже.

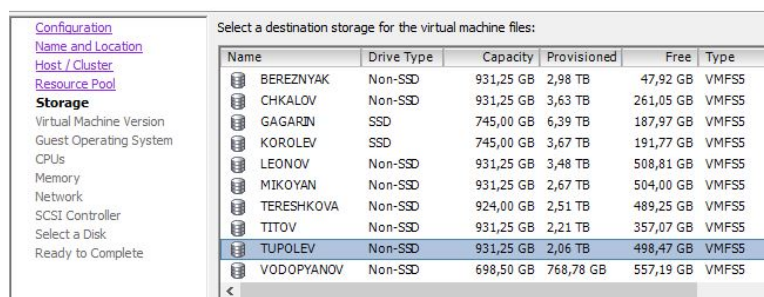


Рисунок 8. Выбор хранилища данных

Следующий шаг заключается в выборе операционной системы, которая позже будет устанавливаться на *ВМ*. Для нашей виртуальной машины будет использоваться ОС *Windows* 10. После этого необходимо выбрать количество процессоров. (Рис. 9)

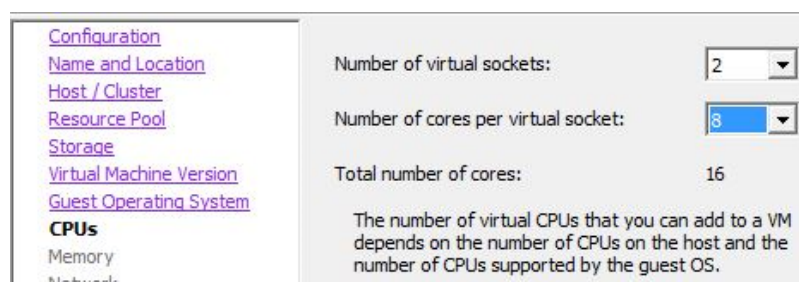


Рисунок 9. Выбор количества процессоров

Затем нужно определить объем оперативной памяти для *ВМ*. (Рис. 10) Как и в случае с процессорами, объем памяти зависит от объема физической памяти хоста. Мастер настройки *ВМ* предоставляет рекомендации по максимальному и минимальному объему используемой памяти, а также рекомендации по обеспечению оптимальной производительности.

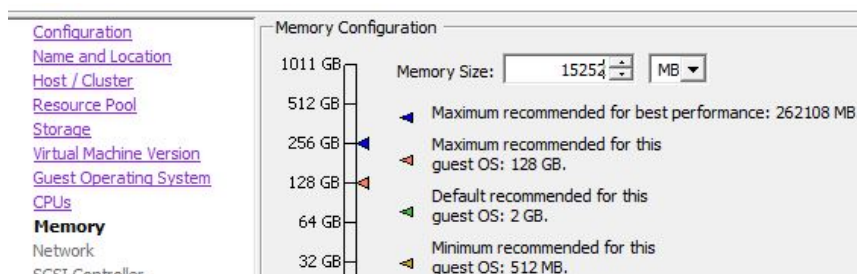


Рисунок 10. Выбор объема оперативной памяти

После этого необходимо выбрать сетевые карты, которые будут использоваться виртуальной машиной. (Рис. 11)

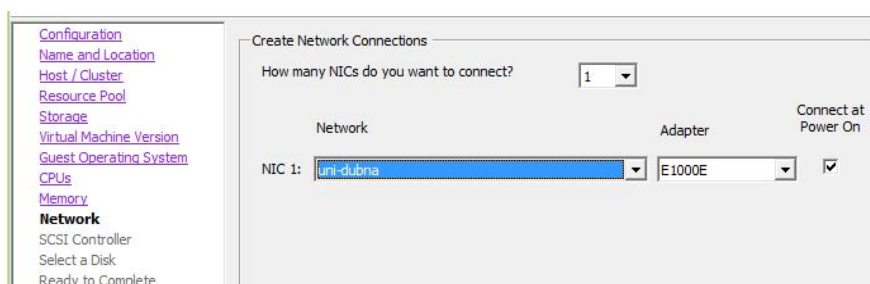


Рисунок 11. Выбор сетевой карты

Следующий шаг заключается в создании виртуального жесткого диска и выборе его объема. (Рис. 12) Размер зависит от объема пространства выбранного ранее хранилища.

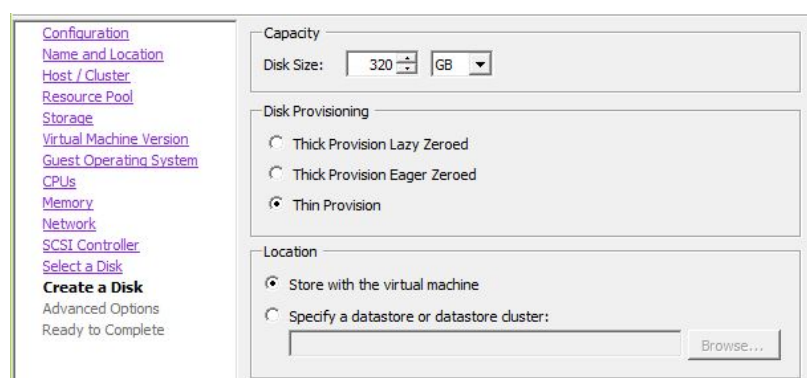


Рисунок 12. Создание виртуального жесткого диска для ВМ

Виртуальная машина успешно создана. Теперь можно посмотреть ее настройки. Для этого необходимо щелкнуть правой кнопкой мыши по ВМ и выбрать *Edit Settings*. Полученные настройки представлены на рис ниже.

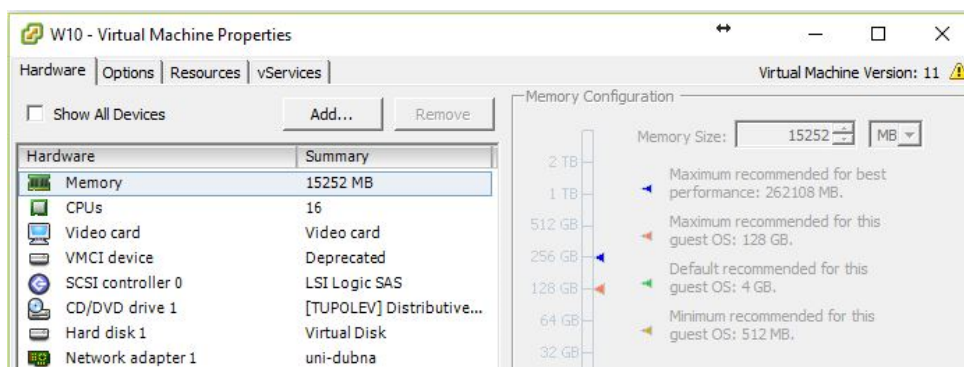


Рисунок 13. Настройки созданной VM

Теперь можно приступить к установке операционной системы. Она выполняется точно так же, как и на обычные ПК, но имеются различные способы применения установочного носителя с ОС. Для установки *Windows 10* на виртуальную машину будет использоваться образ *ISO*, который будет смонтирован в виртуальном *CD/DVD* приводе. Для этого в настройках *VM* необходимо выбрать из списка *CD/DVD drive*, щелкнуть по *Datastore ISO file* и выбрать необходимой дистрибутив. (Рис. 14)

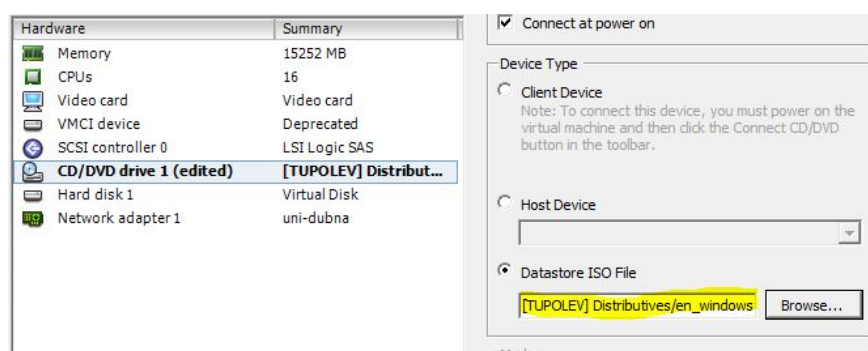


Рисунок 14. Загрузка *ISO* дистрибутива в виртуальный *CD/DVD* привод

После этого при включении *VM* начнется установка ОС *Windows 10*. Процесс установки прост и не требует описания. Теперь можно приступить к разворачиванию программно-инструментальных средств.

### 3.2. Развертывание программно-инструментальных средств

После создания виртуальной машины и установки ОС *Windows 10* можно перейти к следующему этапу – развертыванию необходимого ПО.

Для написания приложений на языке *C#* необходима среда разработки *Visual Studio*. Для того, чтобы ее установить, необходимо перейти в *Microsoft Imagine* и в магазине университета «Дубна» выбрать продукт *Visual Studio 2017 Enterprise* и скачать установочный файл. (Рис. 15)

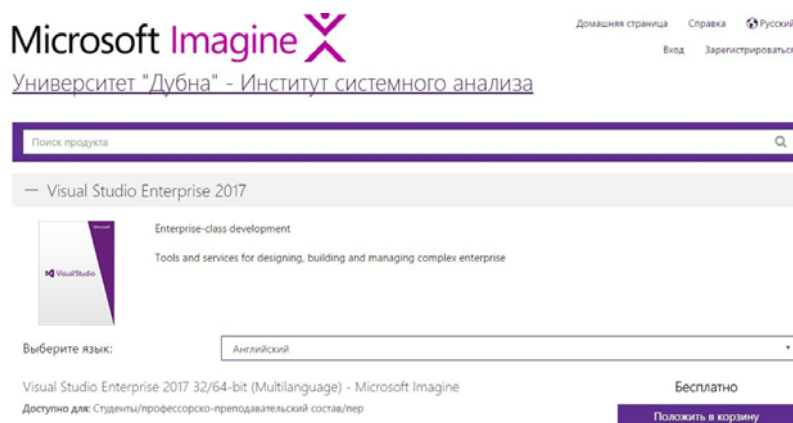


Рисунок 15. Покупка продукта *Visual Studio 2017 Enterprise*

После этого, запустив установочный файл, согласиться с лицензионным соглашением. Для этого нажать кнопку *Continue*. Выбрать компоненты, которые необходимо установить. (Рис. 16) Дождаться завершения процесса установки, и среда разработки *Visual Studio 2017 Enterprise* готова к работе.

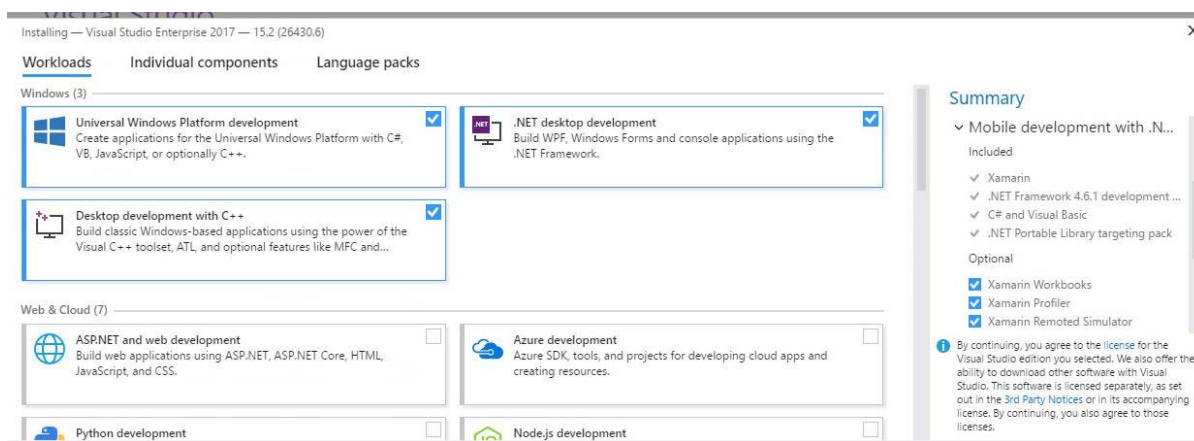


Рисунок 16. Выбор компонентов для установки в *Visual Studio*

Так как у студентов САУ есть курс по разработке *Android* приложений на языке программирования *Java*, в качестве лабораторной работы они смогут сами установить *Android Studio* для того, чтобы создавать мобильные приложения, использующие когнитивные облачные сервисы.



### 3.3. Разработка демонстрационного набора программ

#### Основные понятия

Токен – это ключ для доступа к службам, который выдается пользователю после успешной авторизации или подписки на службу, используется для безопасного удаленного доступа к ресурсам.

*NuGet* – система управления пакетами для платформ разработки *Microsoft*.

#### 3.3.1. *Microsoft Cognitive Services*

##### Получение token для работы с API

Чтобы создавать приложения, использующие преимущества *Cognitive Services*, необходимо подписаться на интересующий сервис и получить секретный ключ (токен).

Для получения бесплатного доступа к пробным подпискам *API Microsoft Cognitive Services* нужно перейти на сайт <https://www.microsoft.com/cognitive-services/> и выбрать облачный сервис, который необходимо активировать. (Рис. 17)

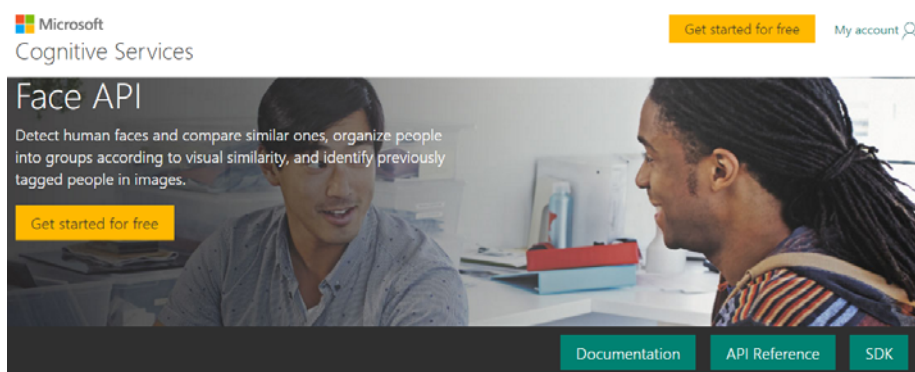


Рисунок 17. *Face API* на сайте *Microsoft Cognitive Services*

Затем нужно кликнуть по кнопке «*Get started for free*» и авторизоваться с помощью своей учетной записи *Microsoft*. После этого необходимо нажать на кнопку «*Subscribe to new free trials*» (зеленый плюс), выбрать из списка службы, которые будут использоваться, прочитать пользовательское соглашение, согласиться с ним, поставив галку рядом с «*I agree to the Microsoft Cognitive Services Terms and Microsoft Privacy Statement*», и нажать кнопку «*Subscribe*». Список служб представлен на рис. ниже (может меняться со временем).

После этого переходим на страницу управления службами. Здесь представлен список доступных сервисов и два ключа доступа к каждому из них. Использовать можно любой ключ, если вдруг он не работает, достаточно нажать на кнопку «*Regenerate*» для выдачи нового токена доступа.

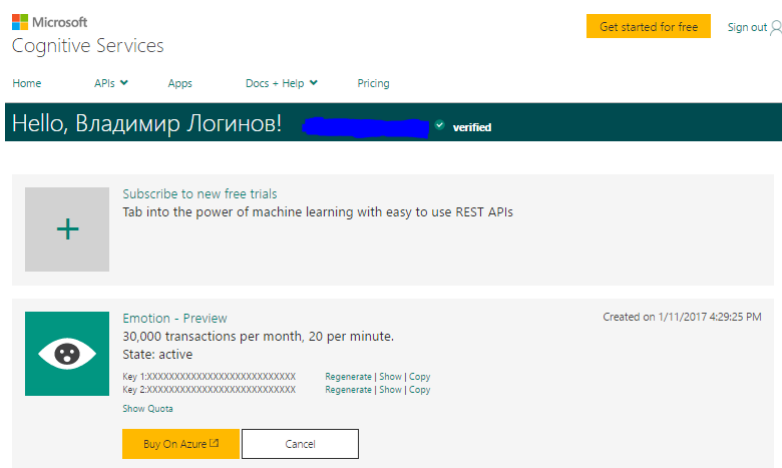


Рисунок 18. Страница управления службами

Если в списке с доступными службами нет необходимого сервиса, значит доступ к нему можно получить через *Microsoft Azure*. Для этого необходимо перейти на сайт <https://portal.azure.com>, войти в учетную запись *Microsoft*, если еще не авторизованы. Для использования служб *Cognitive Services* необходимо оформить подписку с оплатой по мере использования, иначе она будет недоступна для подключения. После этого выбираем в меню раздел «АНАЛИТИКА», в нем щелкаем по пункту *Cognitive Services*. На рис. 19 представлено местонахождение этого раздела.

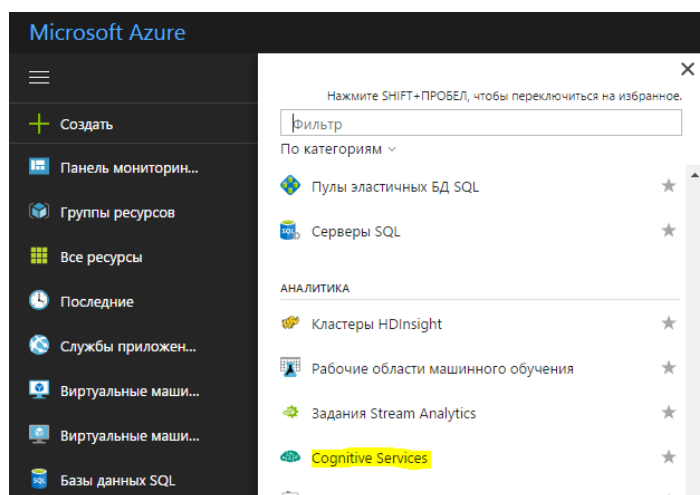


Рисунок 19. Меню на портале *Azure*

Затем жмем кнопку «Добавить подписку» (плюс в верхнем левом углу рабочей области). Заполняем поле «Имя учетной записи», выбираем тип *API* и ценовую категорию, соглашаемся с пользовательскими условиями и нажимаем кнопку «Создать». Пример заполнения представлен на рис. ниже.



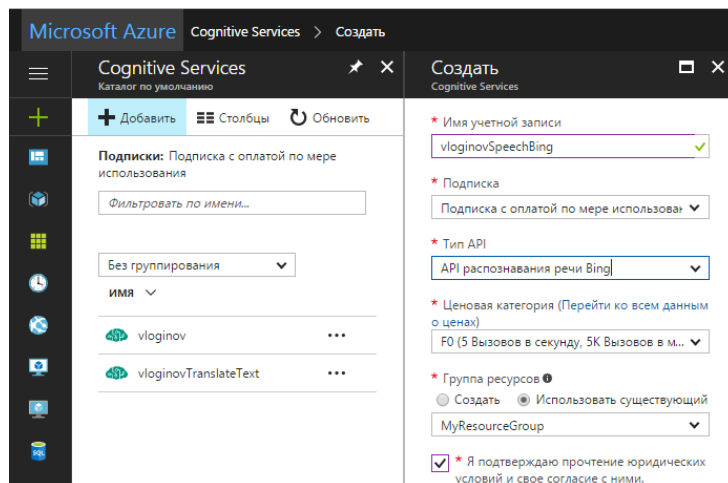


Рисунок 20. Пример заполнения полей при создании подписки на *API Cognitive Services*

Подписка на выбранный *API* сервис успешно создана. Для того, чтобы узнать токен доступа, необходимо щелкнуть по названию сервиса и перейти во вкладку ключи. Пример можно увидеть на рис. 21.

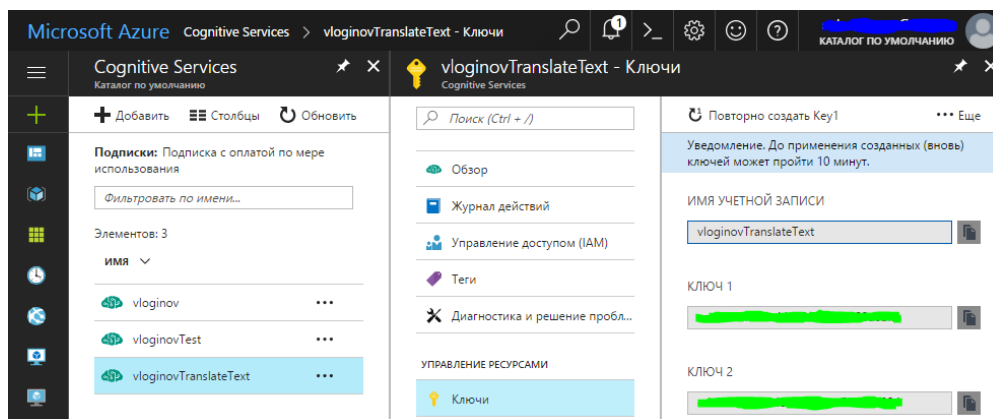


Рисунок 21. Ключи доступа к подписке *API* в *Microsoft Azure*

## Computer Vision

*Computer Vision API* позволяет анализировать и описывать изображения, используя естественный язык. Можно передать локальное изображение сервису или указать *URL* этого изображения и получить обратно полное описание на естественном языке без необходимости в самостоятельном конструировании и форматировании описаний.

Чтобы использовать *Computer Vision API*, необходимо получить секретный ключ для применения в своем коде. Для использования *Computer Vision API* в настольном приложении, запустим среду разработки *Visual Studio* и создадим новый проект *WPF*. (Рис. 22)

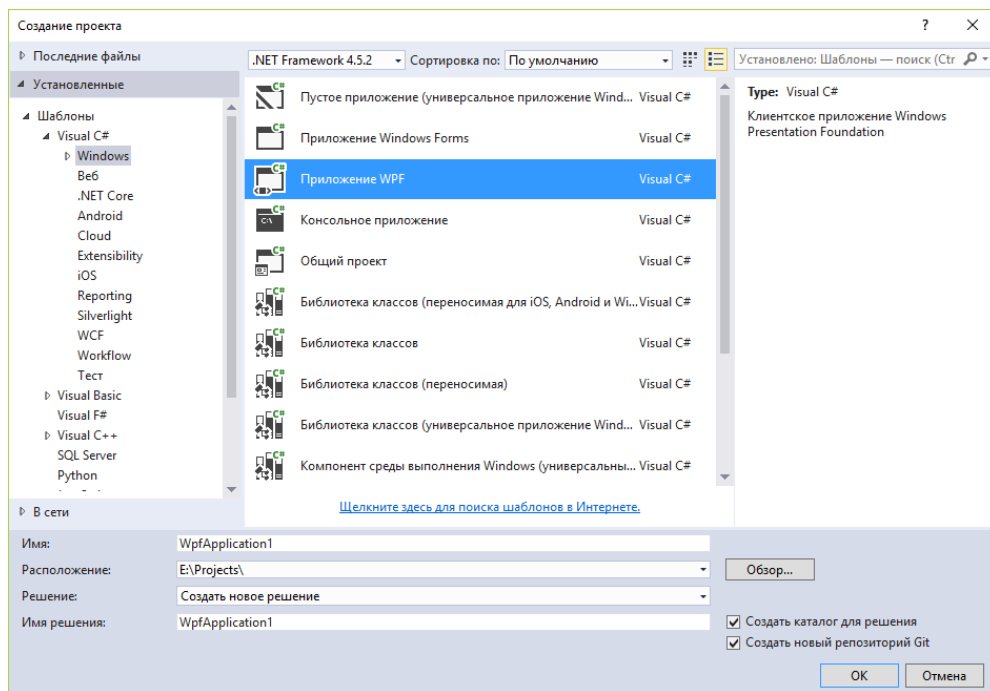


Рисунок 22. Создание *WPF* приложения в среде разработки *Visual Studio* 2017

Для быстрого доступа к службе компьютерного зрения установим *NuGet* пакет «*Microsoft.ProjectOxford.Vision*» (содержит клиентскую библиотеку для *Computer Vision API*). Для этого необходимо щелкнуть правой кнопкой мыши по разделу «Ссылки» и выбрать пункт «Управление *NuGet* пакетами». Затем в строке поиска ввести нужное название и нажать кнопку «Установить».

Создаем два метода для анализа изображения:

- *AnalyzeUrl* – для анализа изображения по *URL* изображения;
- *AnalyzeLocal* – для анализа локального изображения с компьютера.

Для этого используем 3 класса:

- *VisionServiceClient* – основной класс, отвечающий за обработку изображения. (в качестве параметра передается токен доступа);

- *VisualFeature* – класс, отвечающий за характеристики, которые вернет сервер;
- *AnalysisResult* – класс-модель, в которой содержатся результаты анализа изображения.

Пример кода представлен на рис. ниже.

```
private async Task<AnalysisResult> AnalyzeUrl(string url)
{
    VisionServiceClient client = new VisionServiceClient(key);

    VisualFeature[] feature = new VisualFeature[] { VisualFeature.Description, VisualFeature.Tags };
    AnalysisResult result = await client.AnalyzeImageAsync(url, feature);
    return result;
}

private async Task<AnalysisResult> AnalyzeLocal(string path)
{
    VisionServiceClient client = new VisionServiceClient(key);

    using (Stream imageFileStream = File.OpenRead(path))
    {
        VisualFeature[] feature = new VisualFeature[] { VisualFeature.Description, VisualFeature.Tags };
        AnalysisResult result = await client.AnalyzeImageAsync(imageFileStream, feature);
        return result;
    }
}
```

Рисунок 23. Методы для обработки изображений с помощью *Computer Vision API*

Пример программы представлен на рис. 24. В левой части здесь представлено обрабатываемое изображение, а в правой результат анализа, который вернул облачный сервис *Microsoft*.

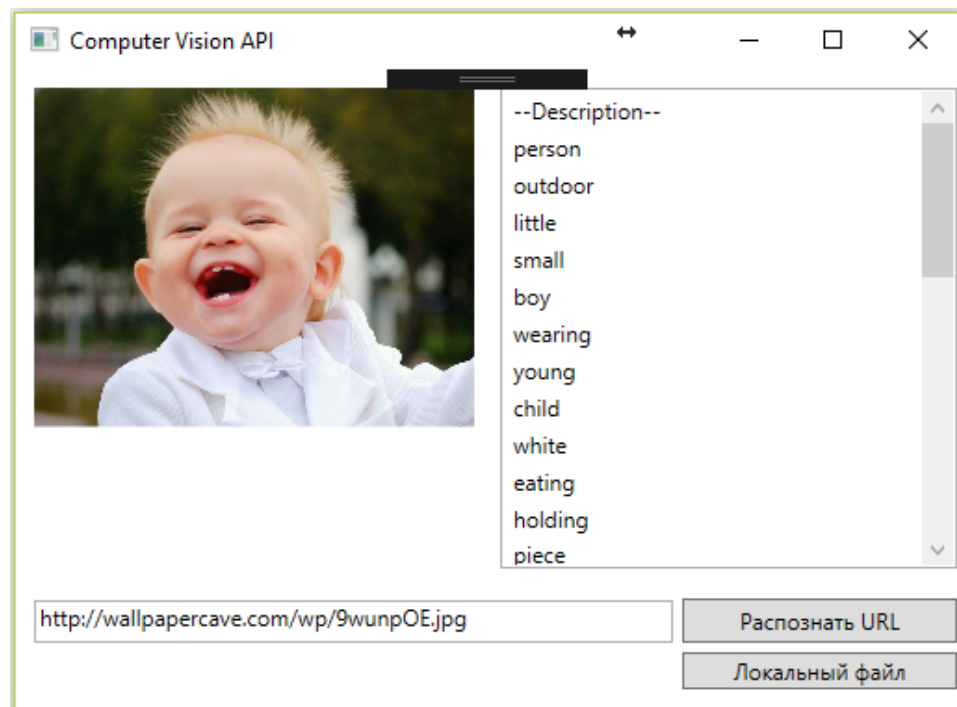


Рисунок 24. Пример использования *Computer Vision API* в *WPF* приложении

## Emotion API

*Emotion API* позволяет анализировать изображения и определять эмоции. Поддерживается распознавание 8 эмоций: гнева, презрения, отвращения, страха, радости, спокойствия, грусти и удивления. Результат может варьироваться от 0 до 1.

Чтобы использовать *Emotion API* необходимо получить токен доступа. Для использования *Emotion API* в настольном приложении, запустим среду разработки *Visual Studio* и создадим новый проект *WPF*.

Для быстрого доступа к службе распознавания эмоций установим *NuGet* пакет «*Microsoft.ProjectOxford.Emotion*». Для этого необходимо щелкнуть правой кнопкой мыши по разделу «Ссылки» и выбрать пункт «Управление *NuGet* пакетами». Затем в строке поиска ввести нужное название и нажать кнопку «Установить».

Для доступа к сервису необходимо использовать два класса:

- *EmotionServiceClient* – класс, отвечающий за доступ к сервису *Emotion API* (в качестве параметра передается секретный ключ);
- *Emotion* – класс-модель, в которой содержатся результаты анализа изображения.

Создаем два метода для анализа эмоций на изображении:

- *AnalyzeEmotionURL* – для анализа изображения по *URL* из интернета;
- *AnalyzeEmotionLocal* – для анализа локального изображения с компьютера.

Пример кода представлен на рис. ниже.

```
private async Task<Emotion []> AnalyzeEmotionURL(string url)
{
    EmotionServiceClient emotionServiceClient = new EmotionServiceClient(key);
    Emotion[] emotionResult = await emotionServiceClient.RecognizeAsync(url);
    return emotionResult;
}

private async Task<Emotion []> AnalyzeEmotionLocal(string path)
{
    EmotionServiceClient emotionServiceClient = new EmotionServiceClient(key);

    using (var imageFileStream = File.OpenRead(path))
    {
        Emotion[] emotionResult = await emotionServiceClient.RecognizeAsync(imageFileStream);
        return emotionResult;
    }
}
```

Рисунок 25. Методы для доступа к сервису *Emotion API*

Для того, чтобы название эмоций выводились на русском языке, а их вероятность была от 0 до 100% напишем метод *Parser*. Код этого метода представлен на рис. 26.

```

private void Parser(Emotion[] emotion)
{
    listBox.Items.Clear();

    if (emotion.Count() > 0)
    {
        listBox.Items.Add("гнев: " + Convert.ToInt32(emotion[0].Scores.Anger * 100));
        listBox.Items.Add("презрение: " + Convert.ToInt32(emotion[0].Scores.Contempt * 100));
        listBox.Items.Add("отвращение: " + Convert.ToInt32(emotion[0].Scores.Disgust * 100));
        listBox.Items.Add("страх: " + Convert.ToInt32(emotion[0].Scores.Fear * 100));
        listBox.Items.Add("счастье: " + Convert.ToInt32(emotion[0].Scores.Happiness * 100));
        listBox.Items.Add("безразличие: " + Convert.ToInt32(emotion[0].Scores.Neutral * 100));
        listBox.Items.Add("печаль: " + Convert.ToInt32(emotion[0].Scores.Sadness * 100));
        listBox.Items.Add("удивление: " + Convert.ToInt32(emotion[0].Scores.Surprise * 100));
    }
    else
        listBox.Items.Add("Не найдено эмоций на фото");
}

```

Рисунок 26. Метод *Parser* для вывода информации о полученных эмоциях

Пример программы представлен на рис. ниже. В левой части здесь представлено обрабатываемое изображение, а в правой результат анализа изображения на наличие эмоций.

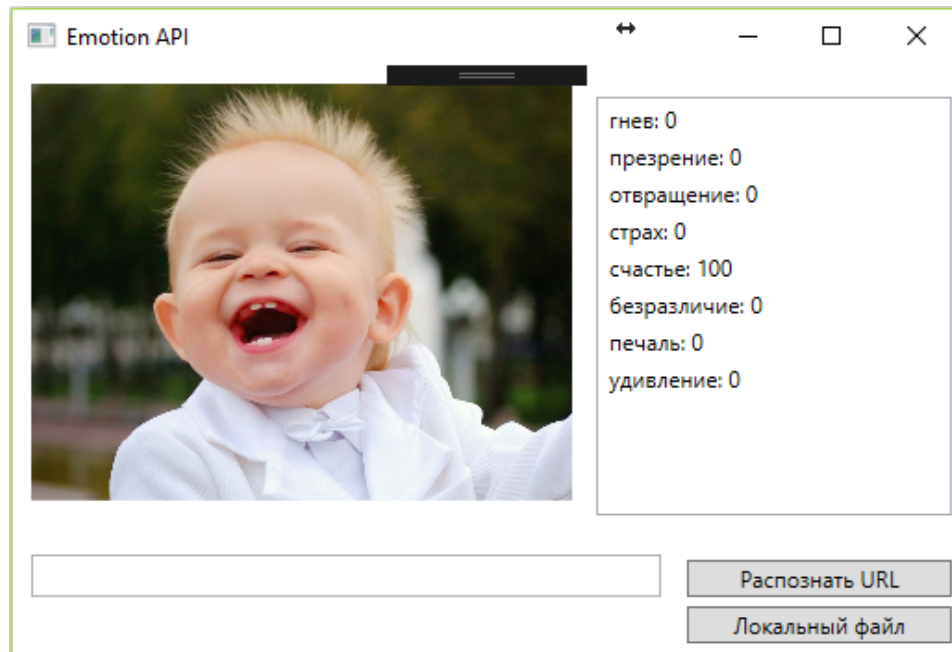


Рисунок 27. Пример использования *Emotion API* в *WPF* приложении

## *Face API*

Набор служб *Face API* включает в себя облачные алгоритмы обнаружения и распознавания лиц на фотографиях, а именно:

- Обнаружение границ лиц в виде описывающих прямоугольников с выделением таких характеристик, как определение координат частей лица, пола и оценка возраста;
- Оценка схожести двух лиц, поиск похожих лиц на серии фотографий по образцу, распознавание людей на основе обучающей выборки.

Сервису можно передать *URL* изображения либо локальное изображение и получить в ответ полное описание в виде *JSON* объекта. Чтобы использовать службы *Face API* необходимо получить токен доступа в *Azure* (для получения пробной подписки нужно воспользоваться сервисом <https://www.microsoft.com/cognitive-services/>). Для использования *Face API* в настольном приложении, запустим среду разработки *Visual Studio* и создадим новый проект *WPF*.

Для быстрого доступа к службе распознавания лиц установим *NuGet* пакет «*Microsoft.ProjectOxford.Face*». Для этого необходимо щелкнуть правой кнопкой мыши по разделу «Ссылки» и выбрать пункт «Управление *NuGet* пакетами». Затем в строке поиска ввести нужное название и нажать кнопку «Установить».

Для доступа к сервису необходимо использовать два класса:

- *FaceServiceClient* – класс, отвечающий за доступ к сервису *Face API* (в качестве параметра передается секретный ключ);
- *Face* – класс-модель, в которой содержатся результаты анализа изображения.

Создаем два метода для анализа лиц на изображении:

- *UploadAndDetectFacesURL* – для анализа изображения по *URL* из интернета;
- *UploadAndDetectFacesLocal* – для анализа локального изображения с компьютера.

Пример кода представлен на рис. ниже.

```

private async Task<FaceRectangle[]> UploadAndDetectFacesURL(string url)
{
    var faceServiceClient = new FaceServiceClient(key);
    var faces = await faceServiceClient.DetectAsync(url);
    var faceRect = faces.Select(face => face.FaceRectangle).ToArray();
    return faceRect;
}

private async Task<FaceRectangle[]> UploadAndDetectFacesLocal(string path)
{
    using (Stream imageFileStream = File.OpenRead(path))
    {
        var faceServiceClient = new FaceServiceClient(key);
        var faces = await faceServiceClient.DetectAsync(imageFileStream);
        var faceRect = faces.Select(face => face.FaceRectangle).ToArray();
        return faceRect;
    }
}

```

Рисунок 28. Методы для доступа к службе *Face API*

Для отрисовки границ лиц в виде описывающих прямоугольников на фотографии создаем метод *Draw*. В качестве параметров принимаются обрабатываемое изображение и результат выполнения метода анализа лиц. Пример кода представлен на рис. 29.

```

private void Draw(BitmapImage bitmapSource, FaceRectangle[] faceRect)
{
    if (faceRect.Count() > 0)
    {
        DrawingVisual visual = new DrawingVisual();
        DrawingContext drawingContext = visual.RenderOpen();
        drawingContext.DrawImage(bitmapSource, new Rect(0, 0, bitmapSource.Width, bitmapSource.Height));
        double dpi = bitmapSource.DpiX;
        double resizeFactor = 96 / dpi;

        foreach (var item in faceRect)
        {
            drawingContext.DrawRectangle(Brushes.Transparent, new Pen(Brushes.Red, 2),
                new Rect(item.Left * resizeFactor, item.Top * resizeFactor, item.Width * resizeFactor, item.Height * resizeFactor));
        }

        drawingContext.Close();
        RenderTargetBitmap faceWithrectBitmap = new RenderTargetBitmap(
            (int)(bitmapSource.PixelWidth * resizeFactor),
            (int)(bitmapSource.PixelHeight * resizeFactor), 96, 96, PixelFormats.Pbgra32);

        faceWithrectBitmap.Render(visual);
        image.Source = faceWithrectBitmap;
    }
}

```

Рисунок 29. Метод отрисовки лиц на изображении

Результат работы программы представлен на рис. ниже.



Рисунок 30. Пример использования *Face API* в *WPF* приложении

## Text Analytics API

*Text Analytics API* позволяет превратить неструктурированный текст в важные сведения. С помощью этого сервиса можно легко определить мнение, выделить ключевые фразы и тему, определить язык для текста любого рода.

Чтобы использовать *Text Analytics API* необходимо получить токен доступа. Для применения этой службы в настольном приложении, запустим среду разработки *Visual Studio* и создадим новый проект *WPF*.

Для использования сервиса необходимо получить токен. Для применения *Text Analytics API* в настольном приложении, запустим среду разработки *Visual Studio* и создадим новый проект *WPF*. Так как готового *Nuget* пакета, на момент написания этой магистерской диссертации, не существует, воспользуемся стандартной *Net* библиотекой *System.Net.Http* для написания асинхронных *Rest* запросов.

Использоваться будут две службы анализа текста:

- *Detect Languages* – для определения языка текста;
- *Detect Phrases* – для выделения ключевых фраз.

Для применения этих служб в приложении были написаны два метода, которые представлены на рис. ниже.

```
private async Task<string> DetectLanguages(int countLanguages, string text)
{
    string result = String.Empty;
    using (var client = new HttpClient())
    {
        client.DefaultRequestHeaders.Add("Ocp-Apim-Subscription-Key", key);
        var uri = "https://westus.api.cognitive.microsoft.com/textanalytics/v2.0/languages?numberOfLanguagesToDetect=" + countLanguages;
        byte[] byteData = Encoding.UTF8.GetBytes("{\"documents\": [{\"id\": \"string\", \"text\": \"\" + text + \"\"}]}");

        using (var content = new ByteArrayContent(byteData))
        {
            content.Headers.ContentType = new System.Net.Http.Headers.MediaTypeHeaderValue("application/json");
            var response = await client.PostAsync(uri, content);

            if (response.IsSuccessStatusCode) result = await response.Content.ReadAsStringAsync();
            else result = response.StatusCode.ToString();
        }
    }
    return result;
}

private async Task<string> DetectKeyPhrases(string language, string text)
{
    string result = String.Empty;
    using (var client = new HttpClient())
    {
        client.DefaultRequestHeaders.Add("Ocp-Apim-Subscription-Key", key);
        var uri = "https://westus.api.cognitive.microsoft.com/textanalytics/v2.0/keyPhrases";
        byte[] byteData = Encoding.UTF8.GetBytes("{\"language\": \"\" + language + \"\", \"id\": \"string\", \"text\": \"\" + text + \"\"}");

        using (var content = new ByteArrayContent(byteData))
        {
            content.Headers.ContentType = new System.Net.Http.Headers.MediaTypeHeaderValue("application/json");
            var response = await client.PostAsync(uri, content);

            if (response.IsSuccessStatusCode) result = await response.Content.ReadAsStringAsync();
            else result = response.StatusCode.ToString();
        }
    }
    return result;
}
```

Рисунок 31. Методы доступа к *Text Analytics API*



На рис. 32. Представлен результат выполнения метода определения языка текста (*Detect Languages*). Слева здесь находится текст для анализа, а справа результат работы *API*.

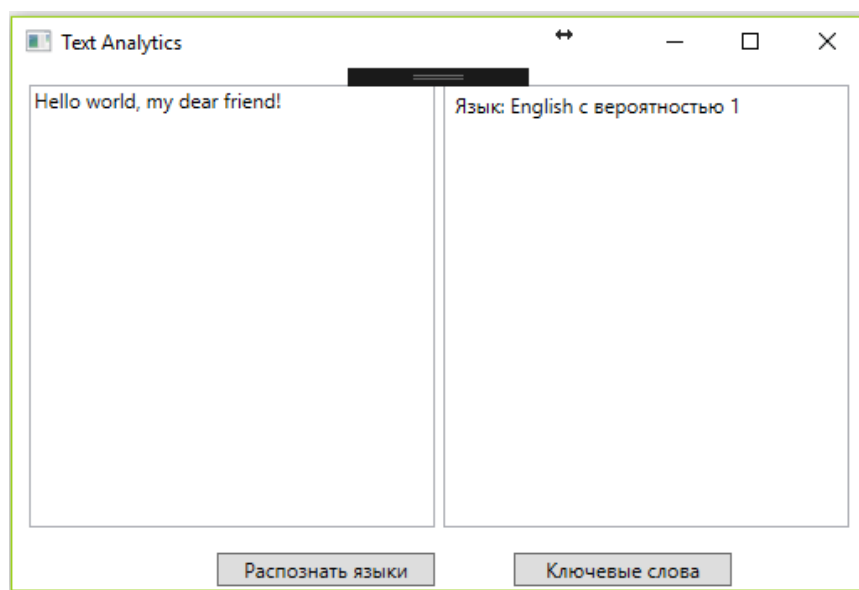


Рисунок 32. Результат выполнения метода *Detect Languages*

Ниже на рис. 33 представлен результат выполнения метода определения ключевых фраз в тексте (*Detect Phrases*). Слева здесь текст для анализа, а справа результат работы метода *API*.

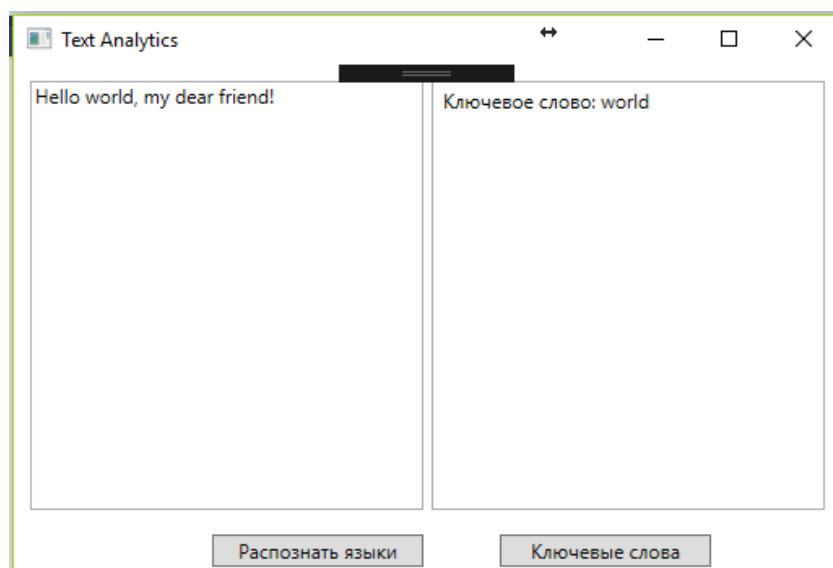


Рисунок 33. Результат выполнения метода *Detect Phrases*

### 3.3.2. IBM Watson API

#### Получение token для работы с API

Для того чтобы получить токена доступа к сервисам *IBM Watson*, необходимо перейти на сайт <https://www.ibm.com/watson/developercloud/>, авторизоваться, если уже есть учетная запись или зарегистрироваться, если ее нет (*IBM* предоставляет бесплатную 30-дневную подписку на *Bluemix*).

После авторизации щелкаем по выпадающему меню (три полоски в левом верхнем углу) и выбираем пункт «*Services*» и подпункт «*Watson*» как на рис. ниже.

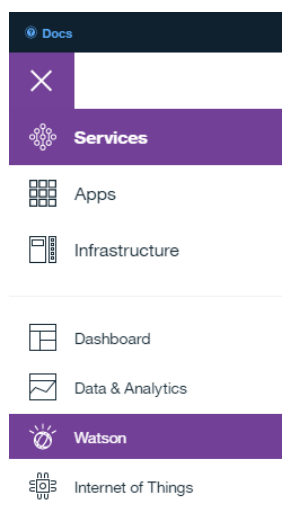


Рисунок 34. Меню *IBM Bluemix*

После этого нажимаем «*Create Watson service*». (Рис. 35)

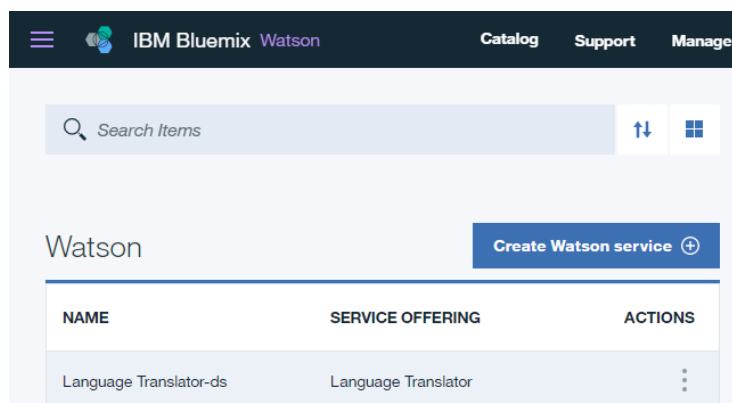



Рисунок 35. Страница *Watson Services*

Выбираем сервис, который будем в дальнейшем использовать и кликаем по нему. После этого выбираем тарифный план и нажимаем кнопку «*Create*». (Рис. 36)

## Pricing Plans

Monthly prices shown are for country or region: [Russian Federation](#)

PLAN	FEATURES	PRICING
 <b>Free</b>	<b>250 Events (images) per day per Bluemix Account</b> 1 Instance per Bluemix Organization	Free
<hr/>		
The Visual Recognition Free Plan offers 250 Events (images) per day per Bluemix Account for image tagging and face detection.		
<b>Standard</b>	<b>Image Tagging Events Pay per Use</b> Face Detection Events Pay per Use Training Events Pay per Use Custom Tagging Events Pay per Use	\$0.00214 USD/Event \$0.00428 USD/Event \$0.107 USD/Event \$0.00428 USD/Event

[Terms](#)

st

Create

Рисунок 36. Тарифные планы подписки

После этого попадаем на страницу службы. Здесь можно попробовать в демо режиме этот сервис, прочитать официальную документацию о нем, а так же в разделе «*Service credentials*» узнать *URL* по которому необходимо выполнять запрос к службе, а так же токен доступа, которые необходим для авторизации. (Рис. 37)

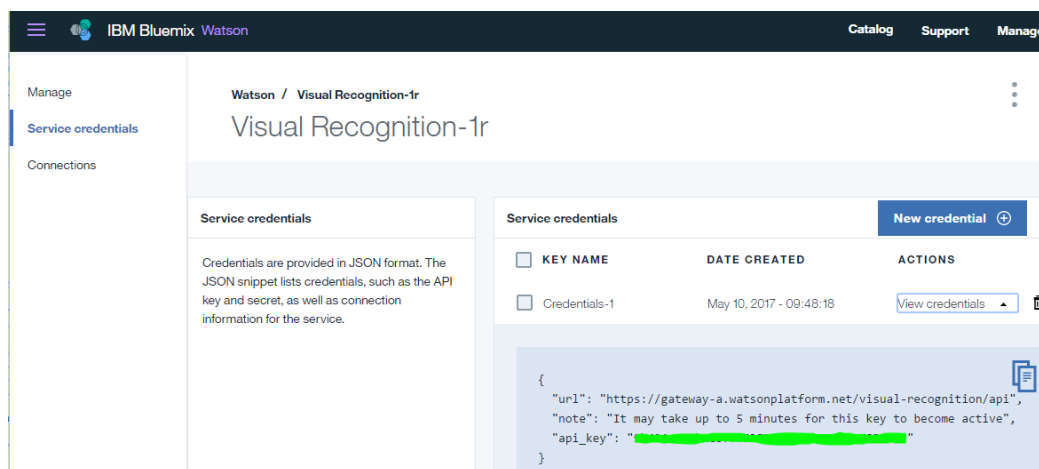


Рисунок 37. Раздел *Service credentials* службы *Visual Recognition*

## Visual Recognition

*Visual Recognition* позволяет распознавать и классифицировать изображения. Кроме того, отличительной особенностью этой службы является то, что можно создавать индивидуальный классификатор изображений, обучая его на собственном материале.

Для использования сервиса необходимо получить токен. Для применения *Visual Recognition API* в настольном приложении, запустим среду разработки *Visual Studio* и создадим новый проект *WPF*. Так как готового *Nuget* пакета, на момент написания этой магистерской диссертации, не существует, воспользуемся стандартной *Net* библиотекой *System.Net.Http* для написания асинхронных *Rest* запросов.

Использоваться будут две службы *Visual Recognition*:

- *Classify* – для распознавания того, что находится на изображении;
- *Detect Faces* – для распознавания лиц на фото, а так же пола и возраста.

Для применения этих служб в приложении были написаны два метода, которые представлены на рис. ниже.

```
private async Task<string> Classify(string api_key, string urlImage)
{
    var result = String.Empty;

    using (var client = new HttpClient())
    {
        client.BaseAddress = new Uri(url);
        var response = await client.GetAsync("/visual-recognition/api/v3/classify?api_key=" + api_key + "&url=" + urlImage + "&version=2016-05-19");
        result = await response.Content.ReadAsStringAsync();
    }
    return result;
}

private async Task<string> Faces(string api_key, string urlImage)
{
    var result = String.Empty;

    using (var client = new HttpClient())
    {
        client.BaseAddress = new Uri(url);
        var response = await client.GetAsync("/visual-recognition/api/v3/detect_faces?api_key=" + api_key + "&url=" + urlImage + "&version=2016-05-19");
        result = await response.Content.ReadAsStringAsync();
    }
    return result;
}
```

Рисунок 38. Методы для доступа к сервису *Visual Recognition*

Эти методы максимально упрощены для лучшего понимания того, как они работают. В качестве параметров методы принимают токен доступа (*api key*) и ссылку на изображение, которое необходимо обработать (*url image*). В качестве ответа они возвращают *JSON* объект с результатом анализа.

Для отрисовки границ лиц на изображении используется метод *Draw*. Справа от изображения выведены пол и возраст людей. Ниже представлен результат выполнения метода *Faces*.

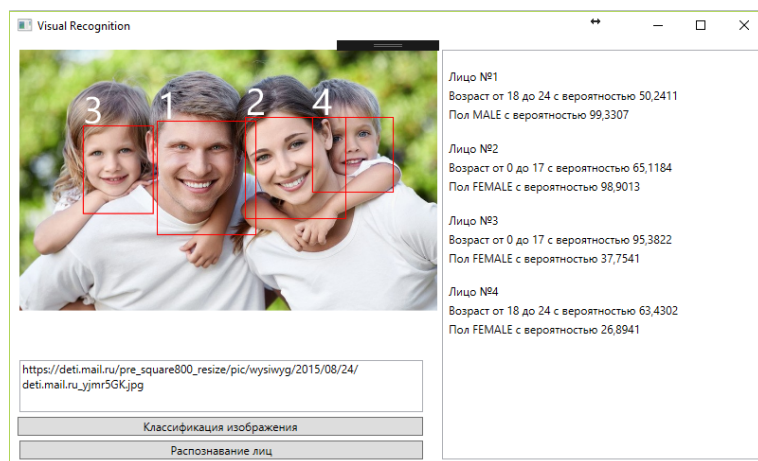


Рисунок 39. Распознавание лиц с помощью *Visual Recognition API*

На рис. 40. представлен результат выполнения метода *Classify*

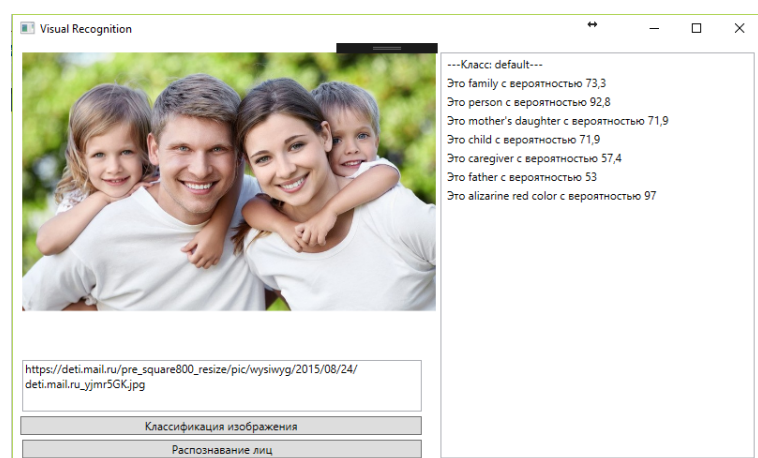


Рисунок 40. Классификация изображения с помощью *Visual Recognition API*

## Language Translator API

*Language Translator* позволяет переводить тексты с одного языка на другой. На данный момент поддерживаются следующие языки: арабский, бразильский, португальский, французский, немецкий, итальянский, испанский и английский.

Для использования *Language Translator API* необходимо получить логин и пароль авторизации на сайте *IBM Watson*. Для применения этого сервиса в настольном приложении, запустим среду разработки *Visual Studio* и создадим новый проект *WPF*. Воспользуемся стандартной *Net* библиотекой *System.Net.Http* для написания асинхронного *GET* запроса.

Для использования *Language Translator API* в приложении, был написан метод *Translate*, который представлен на рис. ниже. В качестве параметров он принимает язык, с которого необходимо выполнить перевод (переменная *source*), язык, на который необходимо перевести текст (переменная *target*), а также текст, который необходимо обработать.

```
private async Task<string> Translate(string _source, string _target, string _text)
{
    string result = String.Empty;

    using (var client = new HttpClient())
    {
        client.BaseAddress = new Uri(url);

        var encoded = Convert.ToBase64String(Encoding.GetEncoding("ISO-8859-1").GetBytes(username + ":" + password));
        client.DefaultRequestHeaders.Add("Authorization", "Basic " + encoded);

        var response = await client.GetAsync("?source=" + _source + "&target=" + _target + "&text=" + _text);
        result = await response.Content.ReadAsStringAsync();
    }
    return result;
}
```

Рисунок 41. Метод доступа к службе *Translate API*

Результат работы программы, можно увидеть на рис. 42. Слева здесь текст, который необходимо перевести (на английском), а справа полученный результат (на испанском).

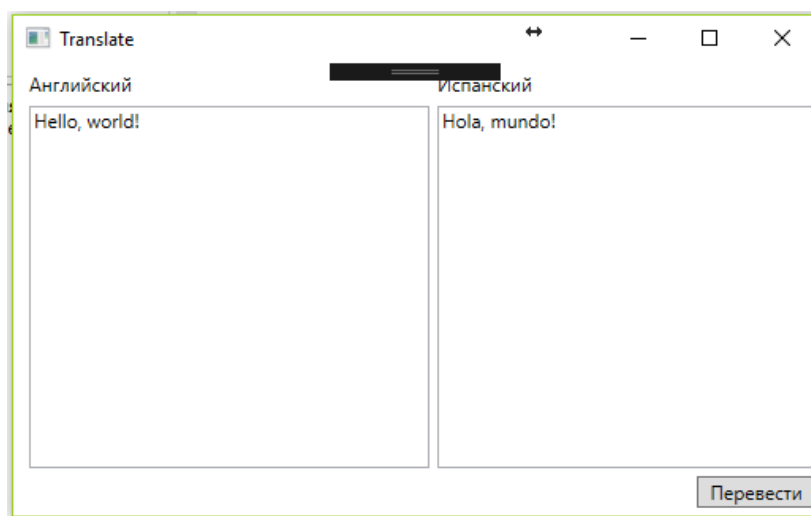


Рисунок 42. Перевод текста с помощью *Translate API*

## Personality Insights API

*Personality Insights* позволяет извлекать характеристики личности на основании того, как пишет человек. Позволяет глубже понять особенности личности, потребности и ценности людей, чтобы построить психологический портрет человека.

Сервис принимает *JSON*, текст или *HTML* (например соц. сети, электронные письма, блоги), написанные одним человеком. На выходе получается дерево когнитивных и социальных характеристик в формате *JSON* или *CSV*.

Для того, чтобы воспользоваться *Personality Insights API* необходимо получить доступ в виде логина и пароля на сайте *IBM Watson*.

После этого создаем *WPF* приложение и добавляем библиотеки *System.Net.Http* и *System.Net.Http.Headers*. Они понадобятся для написания метода, выполняющего *POST* запрос к сервису.

Для использования *Personality Insights API* в приложении, был написан метод *MakeRequestAnalyze*, который представлен на рис. ниже. В качестве параметра он принимает текст, который необходимо обработать.

```
private async Task<string> MakeRequestAnalyze(string text) // Метод построения психологического портрета по тексту
{
    var result = string.Empty;
    using (var client = new HttpClient())
    {
        client.BaseAddress = new Uri(url);
        var encoded = Convert.ToBase64String(Encoding.GetEncoding("ISO-8859-1").GetBytes(username + ":" + password));
        client.DefaultRequestHeaders.Add("Authorization", "Basic " + encoded);
        client.DefaultRequestHeaders.Accept.Add(new MediaTypeWithQualityHeaderValue("application/json"));

        byte[] byteData = Encoding.UTF8.GetBytes(text);
        using (var content = new ByteArrayContent(byteData))
        {
            content.Headers.ContentType = new MediaTypeHeaderValue("text/plain");
            var response = await client.PostAsync(client.BaseAddress, content);
            result = await response.Content.ReadAsStringAsync();
        }
    }
    return result;
}
```

Рисунок 43. Метод доступа к службе *Personality Insights API*

Результат работы программы, можно увидеть на рис. 44. Слева здесь обрабатываемый текст, а справа полученный результат.

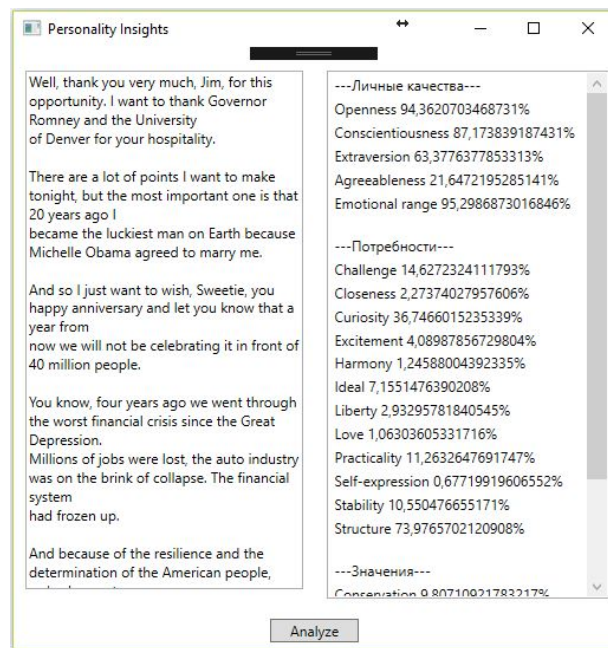


Рисунок 44. Пример использования *Personality Insights API*



### 3.4. Создание видео-инструкций по использованию API когнитивных облачных сервисов

Для лучшего понимания процесса создания приложений, использующих службы облачных когнитивных сервисов были записаны обучающие видеоролики. Для их записи использовалась программа *UVScreenCamera*, которая позволяет выполнять захват видео с экрана, показывает все действия пользователя, включая клики мыши и нажатия сочетаний клавиш, а так же записывает звук с выбранного микрофона. (Рис. 45)

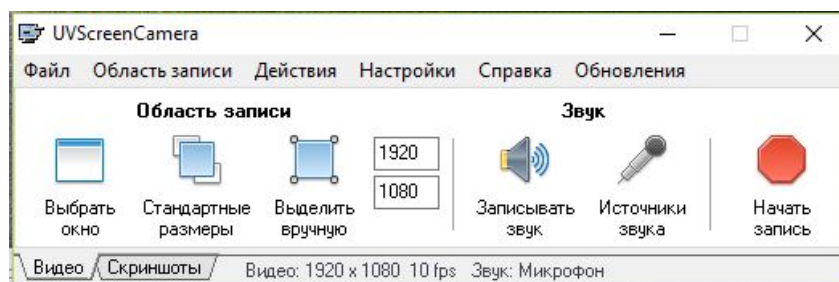


Рисунок 45. Окно программы *UVScreenCamera*

Для того, чтобы студенты могли получить доступ к видео урокам откуда угодно, они были загружены на хостинг *YouTube*. (Рис. 46)

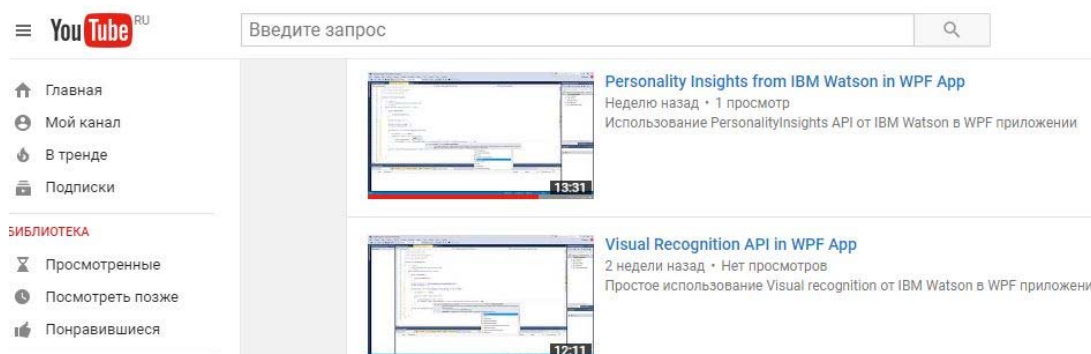


Рисунок 46. Видео-инструкции на хостинге *YouTube*

Позже, все эти инструкции были добавлены к соответствующим урокам на обучающем веб-портале.

### 3.5. Создание обучающего веб-портала

Для создания обучающего веб-портала был выбран конструктор сайтов *wix*. Эта облачная платформа для создания и развития интернет-проектов, позволяет быстро построить сайт на *HTML 5* с помощью инструментов *drag-and-drop*.

Для того, чтобы все обучающие инструкции можно было добавлять как заметки, был выбран шаблон из категории «Блог». После этого выполнены настройки сайта в режиме редактирования. На рис. ниже представлен процесс разработки дизайна.

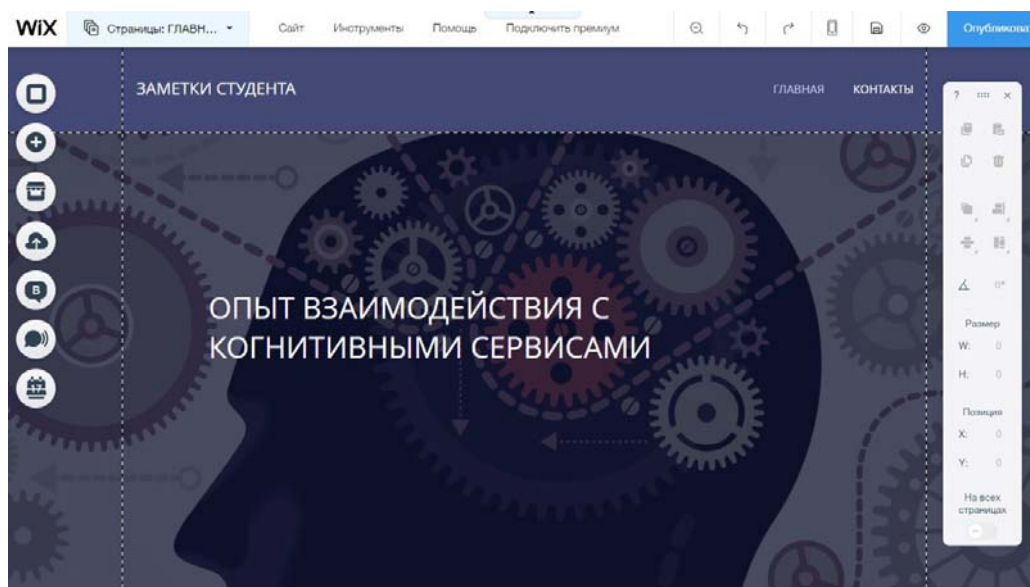


Рисунок 47. Работа над дизайном обучающего портала

Затем, выполнен переход в раздел управления сайтом. (Рис. 48)

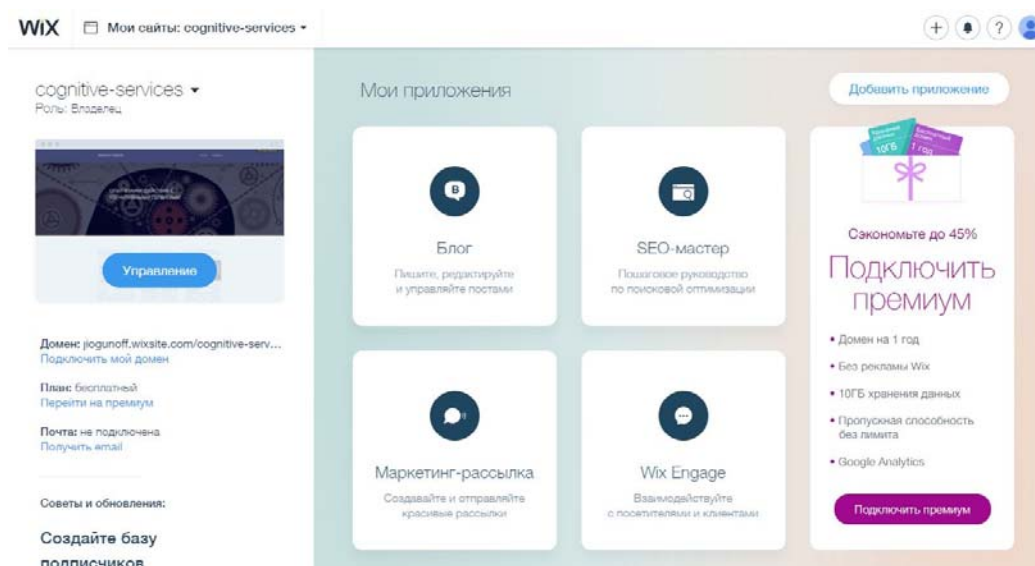


Рисунок 48. Раздел управления сайтом

И перейдя в пункт «Блог», добавлены посты с текстовыми инструкциями по созданию приложений, использующих облачные когнитивные сервисы. Процесс редактирования записи представлен на рис. ниже.

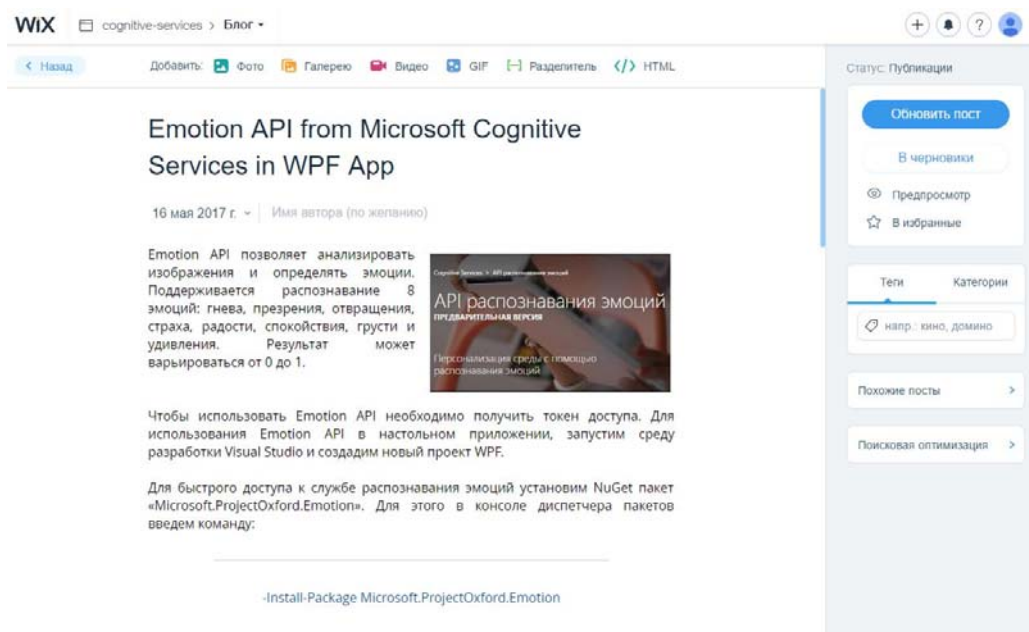


Рисунок 49. Процесс создания поста в блоге

Полученный результат можно увидеть на рис. 50. В каждом посте, помимо текстовой инструкции, прикреплено обучающее видео, процесс создания которого описывался ранее.



Рисунок 50. Обучающий веб-портал на платформе wix

## ЗАКЛЮЧЕНИЕ

В результате проведенных исследований была создана среда освоения когнитивных облачных сервисов в виртуальной компьютерной лаборатории университета «Дубна». Были получены следующие результаты:

- Произведен обзор состояния когнитивных технологий;
- Исследованы методы когнитивной аналитики;
- Выделены классы задач, решаемых с помощью когнитивных технологий;
- Сформированы технологические задачи для которых требуются знания когнитивных технологий;
- Разработан демонстрационный набор программ на языке C#, использующих облачные когнитивные сервисы *Microsoft Cognitive Services* и *IBM Watson*;
- Созданы видео-инструкции по использованию *API* когнитивных сервисов;
- Развернута машина-тренажер в ВКЛ для разработки приложений, использующих «умные» технологии;
- Разработаны образы с предустановленным и настроенным программным обеспечением для выполнения учащимися практических заданий по созданию приложений;
- Создан обучающий веб-портал.

Полученные результаты позволили усовершенствовать образовательный процесс университета «Дубна» и дали возможность студентам *IT*-направлений осваивать когнитивные облачные сервисы для того, чтобы потом их использовать в своих приложениях.

Исследования, проведенные в рамках магистерской диссертации, могут быть продолжены, так как когнитивные технологии, на сегодняшний день, являются одним из приоритетных и постоянно развивающихся направлений в ИТ-сфере.

## СПИСОК ЛИТЕРАТУРЫ

1. Антипов О.Е., Белов М.А., Токарева Н.А. // Архитектура виртуальной компьютерной лаборатории для подготовки специалистов в области информационных технологий. Компьютер в учебном процессе. – Дубна, 2011. – С. 35 – 36.
2. Касаткин П. А. Облачные вычисления – будущее мирового рынка информационных технологий // Научно-методический электронный журнал «Концепт». – 2016. – Т. 34. – С. 138–145. – URL: <http://e-koncept.ru/2016/56752.htm>.
3. Антипов О.Е., Белов М.А., Черемисина Е.Н. // Роль виртуальной компьютерной лаборатории на основе технологии облачных вычислений в современном компьютерном образовании. Научный журнал. – Дубна, 2012. – С. 52 – 62.
4. Антипов О.Е., Белов М.А. // Технология применения виртуальной компьютерной лаборатории в учебных курсах ВУЗа. Естественные и технические науки №1 – Москва, 2012.
5. Антипов О.Е. // Оценка качества инновационной системы поддержки учебного процесса. Журн. «Стандарты и качество» №3 – Москва, 2012.
6. Компания IBM. Когнитивные вычисления – работа быстрее мысли [Электронный ресурс] – Электрон. текст. – 2016. – Режим доступа: <https://habrahabr.ru/company/ibm/blog/276855/>, свободный (дата обращения 10.05.2017).
7. Бахур В. Как когнитивные вычисления и облака решают проблемы анализа больших данных [Электронный ресурс] – Электрон. текст. – 2016. – Режим доступа: [http://ibmstorage.cnews.ru/articles/2016-07-18\\_kak\\_kognitivnye\\_vychisleniya\\_i\\_oblaka\\_reshayut\\_problemy\\_analiza](http://ibmstorage.cnews.ru/articles/2016-07-18_kak_kognitivnye_vychisleniya_i_oblaka_reshayut_problemy_analiza), свободный (дата обращения 11.05.2017)
8. Волоцкий М. Как когнитивные компьютеры могут изменить наше будущее [Электронный ресурс] – Электрон. текст. – 2015. – Режим доступа: <http://gagadget.com/how-it-works/18016-kak-kognitivnyie-kompyuteryi-mogut-izmenit-nashe-budushee/>, свободный (дата обращения 12.05.2017)
9. JJ Geewax. Google Cloud Platform, 2017.
10. Andreas Witting, Michael Witting. Amazon Web Services in Action, 2015.
11. Brian Cooksey. An Introduction to APIs, 2016.
12. John Kelly, Steve Hamm. Smart Machines: IBM's Watson and the Era of Cognitive Computing, 2014.

13. James D Miller. Learning IBM Watson Analytics, 2016.
14. IBM. Cognitive Cooking with Chef Watson: Recipes for Innovation from IBM & the Institute of Culinary Education, 2015.
15. Mitch Tulloch. Introducing Windows Azure for It Professionals, 2013.
16. Leif Larsen. Learning Microsoft Cognitive Services, 2017.
17. IBM Redbooks. IBM Watson Content Analytics: Discovering Actionable Insight from Your Content, 2014.
18. Ситаров В.А. Дидактика: Учеб. пособие для студ. высш. пед. учеб. заведений / Под ред. В.А. Сластенина – М.: Издательский центр «Академия», 2004. С. 60-65.
19. Eric Siegel. Predictive Analytics: The Power to Predict Who Will Click, Buy, Lie, or Die, 2016.
20. Judith Hurwitz, Marcia Kaufman. Cognitive Computing and Big Data Analytics, 2015.