

## **Содержание:**

# **ВВЕДЕНИЕ**

Мы живём в условиях технического прогресса. Развитие современных технологий транспорта, производства и особенно оборонной промышленности происходит ежедневно. Вместе с этим растёт степень автоматизации, компьютеризирование рабочих мест и одновременно возрастает психическая нагрузка на функции операторов. Это происходит во всех сферах оперативной деятельности. Техническое оснащение рабочих мест в разных сферах профессиональной деятельности требует освоения компьютерных технологий, выступающих в качестве орудий труда. В современном мире почти каждый из нас вынужден взаимодействовать с компьютером, смартфоном, терминалами оплат. Именно потому, что эта тема является на сегодняшний момент очень актуальной, я решил раскрыть её в своей работе.

Под понятием «интерфейс» принято понимать набор средств, используемых для взаимодействия двух систем. В переводе с английского слово «interface» буквально означает «место соприкосновения», а под системами, между которыми осуществляется такое взаимодействие, могут подразумеваться различные объекты. Например, это может быть взаимодействие между оборудованием и человеком, различными видами оборудования, но наиболее часто под интерфейсом подразумевают систему взаимодействия программы с человеком для обмена данными и получения нужной информации. Чем удобнее интерфейс, тем меньше он будет являться источником стресса и психологического дискомфорта для пользователя.

На сегодняшний день существует множество вариантов построения интерфейса программ. Возможно, придёт время, когда каждый пользователь среди этого многообразия найдёт для себя удобный и понятный интерфейс.

## **ГЛАВА 1. Виды интерфейсов**

Один из наиболее фундаментальных принципов разработки хорошего программного обеспечения состоит в отделении интерфейса от реализации. Это

облегчает модификацию программ.[\[1\]](#)

### **Элементами интерфейса являются:**

- язык пользователей;
- язык сообщений компьютера, организующий диалог на экране дисплея;
- знания пользователей о правилах работы с системой.

Элементами языка пользователя могут быть формы (шаблоны) входной и выходной информации и т. п.[\[2\]](#)

Рассмотрим наиболее распространённые виды интерфейсов на сегодняшний день.

**1.1. Текстовый пользовательский интерфейс** — разновидность интерфейса пользователя, использующая при вводе-выводе и представлении информации исключительно набор буквенно-цифровых символов и символов псевдографики. Характеризуется малой требовательностью к ресурсам аппаратуры ввода-вывода и высокой скоростью отображения информации. Появился на одном из начальных этапов развития вычислительной техники, при развитии возможностей аппаратуры, нацеленной на реализацию появившегося ранее интерфейса командной строки, который, в свою очередь, является наследником использования телетайпов в качестве интерфейса вычислительной техники. Интерфейс командной строки имеет ряд преимуществ в пользовании перед графическим интерфейсом, поэтому программы с текстовым интерфейсом создаются и используются по сей день, особенно в специфических сферах и на маломощном оборудовании.

Недостатком подобного типа интерфейса является ограниченность изобразительных средств по причине ограниченности количества символов, включённых в состав шрифта, предоставляемого аппаратурой.

Программы с текстовым интерфейсом могут реализовывать оконный интерфейс, чему особенно способствует применение псевдографических символов.

**1. 2. Тактильный интерфейс** - это устройство, позволяющее пользователю взаимодействовать с компьютером посредством получения чувствительной обратной связи. Такие устройства могут использоваться не только людьми с ограниченными возможностями зрения.

**1.3. Жестовый интерфейс** — подмножество системы ввода для графического пользовательского интерфейса для устройств, оснащённых специальными

устройствами ввода или сенсорными экранами, и позволяющее заменять клавиатурные команды при помощи жестов. Основной мотивацией разработки таких интерфейсов является улучшение эргономичности управления, с отказом от привычного для компьютерных программ меню приложения.

Подобный интерфейс может быть реализован как при помощи устройств координатного ввода с возможностью считывания координаты одной точки касания, так и таких, в которых имеется возможность считывания координат более чем одной точки — сенсорные экраны и панели. Последние стали широко применяться в интерфейсах множества современных смартфонов с сенсорным экраном, ноутбуков и прочих мобильных устройств.

В случае устройств с большим размером экрана — например, планшетных ПК, росчерки-жесты являются стандартными функциями интерфейса управления и перьевого ввода. В случае карманных устройств, в отличие от классических графических интерфейсов пользователя, из-за малых физических размеров экрана для произведения росчерка требуется меньшая точность позиционирования, чем для доступа к традиционным элементам графического интерфейса — нажатия «кнопки» или выбора пункта меню.

**1.4. Голосовой интерфейс** при помощи голосовой или речевой платформы делает возможным взаимодействие человека и компьютера для запуска автоматизированного сервиса или процесса.

Ранее контролировать устройство при помощи голоса было возможно только в научной фантастике. До недавнего времени считалось областью искусственного интеллекта. Тем не менее, с развитием технологий ГИ стал всё более распространённым, человек всё чаще пользуется преимуществами этой бесконтактной технологии.

Однако, использование ГИ имеет свои сложности. Люди с малой долей терпения относятся к «машине, которая не понимает». Следовательно, ГИ должен работать почти безошибочно, а значит, стабильно реагировать на входные данные, иначе пользователи не будут применять ГИ, либо ГИ станет объектом насмешек. Для того, чтобы создать стоящий голосовой интерфейс, необходимы междисциплинарные знания в информатике, лингвистике и психологии — дорогостоящие навыки, которые нелегко приобрести. Даже при наличии продвинутых средств разработки, создавая эффективный ГИ, нужно отдавать себе

отчёт в том, какие задачи будет выполнять ГИ, и на какую целевую аудиторию он ориентирован. Чем лучше ГИ будет подходить под когнитивную модель задания пользователя, тем легче будет использовать ГИ без или с минимальным обучением, что повысит его эффективность и степень удовлетворённости пользователей.

Очень важны особенности целевой аудитории. К примеру, создавая ГИ для широкой публики, нужно уделить особое внимание простоте использования и большому количеству инструкций и подсказок для начинающих. В то время, как придумывая ГИ для небольшой группы продвинутых пользователей (в том числе для техподдержки), нужно больше думать о продуктивности, чем о подсказках и инструкциях. Такие приложения должны систематизировать обработку вызовов, минимизировать количество подсказок, устранить ненужные повторения, использовать принцип «смешанной инициативы», который позволит звонящему вносить разные типы информации в одном высказывании и в любом порядке или комбинации. Иными словами, ГИ должен быть специально создан для конкретных технологических процессов, которые нужно автоматизировать.

Голосовой интерфейс подойдёт не для каждого технологического процесса. В общем, чем более многогранны запросы и операции, тем сложнее их будет автоматизировать, и тем большая существует вероятность, что ГИ не подойдет для использования широкой публикой. В некоторых случаях, автоматизация процесса в принципе невозможна, поэтому единственный выход — использование консультанта-человека. К примеру, будет очень сложно автоматизировать горячую линию юридической поддержки. С другой стороны, ГИ прекрасно подойдёт для обработки быстрых и однообразных операций, таких как изменение статуса заказа, заполнение графы «время» или «стоимость», или перевод средств между счетами.

**1.5. Материальный интерфейс пользователя** - это разновидность интерфейса пользователя, в котором взаимодействие человека с электронными устройствами происходит при помощи материальных предметов и конструкций.

Примером материального интерфейса можно назвать шаровой автоответчик Дюрелла Бишопа. Каждый шарик соответствует сообщению, оставленному на автоответчике. Перемещение шарика в специальную выемку воспроизводит связанное с ним сообщение или вызывает звонившего.

Ещё пример - система Топобо. Блоки её напоминают элементы конструктора LEGO, которые могут соединяться вместе, но, в то же время, перемещаться самостоятельно за счёт моторчиков. Можно тянуть, толкать или поворачивать эти элементы; они запомнят эти действия и способны в дальнейшем их воспроизвести.

Другая реализация позволяет пользователю материальным пером набросать рисунок на чувствительном столе МИП-системы. Используя предварительно запрограммированные «жесты», можно клонировать картинку или растягивать по двум осям, как в программе рисования.

## **1.6. Системы с интеллектуальным интерфейсом.**

Интеллектуальные базы данных отличаются от обычных баз данных возможностью выборки по запросу необходимой информации, которая может явно не храниться, а выводиться из имеющейся в базе данных.

Для выполнения первого типа запроса необходимо сначала проведение статистического расчета среднеотраслевой цены по всей базе данных, а уже после этого собственно отбор данных. Для выполнения второго типа запроса необходимо вывести значения характерных признаков объекта, а затем поиск по ним аналогичных объектов. Для третьего типа запроса требуется сначала определить список посредников-продавцов, выполняющих продажу данного товара, а затем провести поиск связанных с ними покупателей.

Во всех перечисленных типах запросов требуется осуществить поиск по условию, которое должно быть доопределено в ходе решения задачи. Интеллектуальная система без помощи пользователя по структуре базы данных сама строит путь доступа к файлам данных. Формулирование запроса осуществляется в диалоге с пользователем, последовательность шагов которого выполняется в максимально удобной для пользователя форме. Запрос к базе данных может формулироваться и с помощью естественно-языкового интерфейса.

**1.7. Естественно-языковой интерфейс** используется для:

- доступа к интеллектуальным базам данных;
- контекстного поиска документальной текстовой информации;
- голосового ввода команд в системах управления;
- машинного перевода с иностранных языков.[\[3\]](#)

Естественно-языковой интерфейс предполагает трансляцию естественно-языковых конструкций на внутримашинный уровень представления знаний. Для этого

необходимо решать задачи морфологического, синтаксического и семантического анализа и синтеза высказываний на естественном языке. Так, морфологический анализ предполагает распознавание и проверку правильности написания слов по словарям, синтаксический контроль - разложение входных сообщений на отдельные компоненты (определение структуры) с проверкой соответствия грамматическим правилам внутреннего представления знаний и выявления недостающих частей и, наконец, семантический анализ - установление смысловой правильности синтаксических конструкций. Синтез высказываний решает обратную задачу преобразования внутреннего представления информации в естественно-языковое.

Простота при работе с естественно-языковым интерфейсом достигается путем применения пользователем языка, используемого в ежедневной коммуникации. Пользователи сети Интернет пытаются найти ответы на вопросы с помощью поисковых машин и форумов. В Интернете уже есть ответы на эти вопросы, вопрос только в том, насколько быстро может быть получен этот ответ.

Для решения подобной проблемы используется естественно-языковой интерфейс в составе диалоговой системы.[\[4\]](#)

**Вывод:** на сегодняшний день есть большое количество интерфейсов программ. Все они разрабатываются для удобства пользователей. Разработчиками учитываются моральные, интеллектуальные и физические особенности людей. Учитываются способности людей с ограниченными возможностями. Например, для людей, которые не могут или не хотят пользоваться клавиатурой и мышью, существует голосовой набор. Примером этому служит жизнь Стивена Хокинга. У него была редкая форма болезни моторных нейронов, которые полностью парализовало его. После потери речи он общался со своим компьютером посредством синтезатора речи. Я считаю, что эволюция интерфейсов происходит для того, чтобы удовлетворить все потребности пользователей.

## **ГЛАВА 2. Особенности построения интерфейса программ**

**2.1. Среда программирования** - это совокупность инструментов, используемых при разработке программного обеспечения. Этот набор обычно состоит из файловой системы, текстового редактора, редактора связей и компилятора.

Дополнительно он может включать большое количество инструментальных комплексов с единообразным интерфейсом пользователя. Старейшей средой программирования считается UNIX - машинно-независимая операционная система с разделением времени. Она предоставляет многочисленные мощные инструментальные средства для производства ПО и эксплуатации разнообразных языков. Работа с этой средой осуществляется с помощью графического интерфейса, устанавливаемого поверх нее. Во многих случаях этим интерфейсом является Common Desktop Environment (CDE). Последнюю стадию развития сред разработки ПО представляют Microsoft Visual C++, Visual BASIC, Delphi и Java Development Kit, которые предлагают легкий способ создания графических интерфейсов для программ пользователя.[\[5\]](#)

Пользовательский интерфейс является своеобразным коммуникационным каналом, по которому осуществляется взаимодействие пользователя и компьютера.

При постоянном использовании интерфейса у вас формируются определенные привычки. Любая привычка означает отказ от внимания к деталям. В этом смысле задача дизайнеров заключается в том, чтобы создавать интерфейсы, которые не позволяют привычкам вызывать проблемы у пользователей. [\[6\]](#)

Самый подходящий пользовательский интерфейс — это такой интерфейс, которому пользователь не должен уделять много внимания, почти не замечать его. Пользователь просто работает, вместо того, чтобы размышлять, какую кнопку нажать или где щелкнуть мышью. Такой интерфейс называют прозрачным — пользователь как бы смотрит сквозь него на свою работу.

Чтобы создать эффективный интерфейс, который делал бы работу с программой приятной, нужно понимать, какие задачи будут решать пользователи с помощью данной программы и какие требования к интерфейсу могут возникнуть у пользователей. Это сделать гораздо легче, если вы используете свою программу для собственных нужд, ведь в данном случае вы являетесь не только разработчиком, но и пользователем программы, смотрите на нее глазами ее аудитории.

Огромную роль играет интуиция — если разработчик сам терпеть не может некрасивые и неудобные интерфейсы, то при создании собственной программы он будет чувствовать, где и какой именно элемент нужно убрать или добавить. «Мы сделали иконки на экране такими хорошенькими, что вам захочется их лизнуть»[\[7\]](#)  
- Писал Стив Джобс в журнале «Fortune»

Если говорить о самых **общих принципах проектирования пользовательских интерфейсов**, то можно назвать три основных положения:

1. Программа должна помогать выполнить задачу, а не становиться этой задачей.
2. При работе с программой пользователь должен чувствовать себя уверенно.
3. Программа должна работать так, чтобы пользователю нравилась работа компьютера.

**2.2. Первый принцип** — это уже упоминавшаяся выше прозрачность интерфейса. Интерфейс должен быть легким для освоения и не создавать перед пользователем преграду, которую он должен будет преодолеть, чтобы приступить к работе. Этот принцип надо исполнять для того, чтобы любой пользователь, даже без предрасположенности пользования системой мог спокойно взаимодействовать с устройством.

**2.3. Второй принцип** часто нарушают те авторы программ, которые слишком недооценивают умственные способности пользователей.

Во-первых, традиционным слегка высокомерным отношением программистов к простым пользователям. Это еще можно было понять в восьмидесятых и начале девяностых годов XX века, когда обычные персональные компьютеры не имели доступных широкой аудитории программных и аппаратных средств для построения привлекательных графических интерфейсов и работы с ними. Самой распространенной операционной системой в то время была MS DOS, основанная на интерфейсе командной строки. Поэтому эффективно работать с персональным компьютером могли люди только с довольно серьезной подготовкой. Кроме того, парк “персоналок” был относительно невелик даже в США, не говоря уже об остальных странах, и, как следствие, число пользователей компьютеров было небольшим.

Сегодня же такой пренебрежительный взгляд на пользователя явно неуместен. Работа с персональным компьютером предполагает относительно не большую начальную подготовку пользователя: интерфейсы компьютерных программ, в первую очередь операционной системы Windows, являющейся законодателем мод в индустрии массового программного обеспечения, становятся все проще и доступнее для понимания людей. Да и число компьютеров в мире сегодня в несколько раз больше, чем десять лет назад.



Вторая причина слишком большой недоверчивости программистов к познаниям и квалификации пользователей – чрезмерное увлечение построением так называемой “защиты от дурака”. Дело в том, что классические учебные курсы по программированию учат, что большинство ошибок в работе программы вызываются не дефектами исходного кода или программного окружения, а действиями пользователя — например, вводом данных неправильного формата (допустим, текста вместо цифр). Поэтому программист при разработке приложения должен написать функции по проверке результатов как можно большего числа действий пользователя и предусмотреть максимальное количество вариантов развития событий. Это совершенно правильный подход, но многие программисты настолько усложняют “защиту от дурака”, делают ее такой громоздкой, что работа пользователя с программой начинает напоминать известное “шаг вправо, шаг влево считается побегом”. Происходит довольно обычная вещь: то, что задумывалось как решение проблемы, само начинает создавать проблемы.

И, наконец, третья причина во многом обусловлена поведением самих пользователей. Часто при возникновении малейших затруднений при работе с программой пользователь тут же обращается в службу технической поддержки, не удосужившись даже взглянуть на справочную систему продукта, посмотреть секцию “Ответы на частые вопросы” на Web-сайте программы или даже просто чуть-чуть подумать! Отчасти тут вина самих авторов программ. Как говорят опытные разработчики пользовательских интерфейсов: “Если уже на этапе знакомства с программой пользователь вынужден обращаться к справочной системе, над интерфейсом нужно серьезно работать”. Поэтому, чтобы соблюсти второй из общих принципов построения интерфейсов и не давать пользователю повода почувствовать, будто его принимают за идиота, не нужно давать разрабатываемой программе слишком большие полномочия и право указывать пользователю, что именно ему делать. Некоторые программисты не знают или не желают осознавать этого и загоняют пользователей своих программных продуктов в тесные рамки, навязывая определенный стиль работы.

Один из примеров такого неправильного отношения к пользователю является отказ программы выполнить вполне естественную с точки зрения пользователя программных продуктов такую рода операцию и вывод диалогового окна, требующего выполнить какую-то другую последовательность действий. Этим “прославился”, например, известный текстовый редактор “Блокнот” из состава Windows 95. Если пользователь открывал эту программу и решал перед началом набора текста дать создаваемому “Блокнотом” по умолчанию файлу “Untitled”

какое-нибудь имя, выбрав из меню команду Сохранить как, редактор отказывался сделать это, показывая сообщение: “Вы не ввели какой-либо текст, чтобы его можно было сохранить. Наберите какой-нибудь текст, а затем попытайтесь [сохранить его] снова”. Этим создатели “Блокнота” не только отвергли стиль работы очень многих пользователей (перед началом набора текста дать имя файла), но сбили с толку и тех, кто был знаком с “Блокнотом” по предыдущим версиям Windows. Например, в шестнадцатизрядной Windows 3.1 “Блокнот” позволял сохранять пустые файлы безо всяких проблем. Опытные пользователи, знакомые с принципами работы операционной системы, тоже были в недоумении: если из контекстного меню Проводника Windows в меню Создать выбрать пункт Текстовый документ, то получившийся файл длиной 0 байт открывается “Блокнотом” без каких-либо затруднений. К счастью, в последующих версиях Windows “Блокнот” стал более дружелюбен к пользователю.

Другой пример недооценки возможностей пользователя — вывод информационных сообщений в ситуациях, когда этого не требуется. Многие авторы наделяют свои программы излишней “болтливостью” из благих намерений — например, для того, чтобы облегчить освоение продукта или информировать пользователей о полезных функциях программы. Однако вполне может оказаться так, что пользователь уже достаточно уверенно чувствует себя при работе с программой и не нуждается в подсказках, выскакивающих каждую минуту, а некоторые полезные, с точки зрения автора программного продукта, функции для конкретного пользователя таковыми не являются. Поэтому среди разработчиков программного обеспечения хорошим тоном считается предоставление пользователю возможности отключить вывод информационных сообщений. Это позволяет сохранить легкость освоения продукта для начинающих пользователей и одновременно с этим добиться, чтобы информационные сообщения не вызывали у опытных пользователей раздражения.

## **2.4. Третий принцип — “Программа должна работать так, чтобы пользователю нравилась работа компьютера”.**

Несмотря на стремительное развитие информационных технологий, многие компьютерные программы все еще имеют примитивный искусственный интеллект. Они прерывают работу пользователя некорректными вопросами и выводят на

экран ненужные сообщения, отвлекая в самых разных ситуациях. В результате многие пользователи, работающие с компьютерами, раздражаются от некорректной работы компьютера.

Пользователь, поглощенный своей задачей, не замечает уведомления, подсказки и другие попытки системы пообщаться с ним.[\[8\]](#)

**Вывод:** Разработчики интерфейса программ должны учитывать желания и возможности огромного количества пользователей. Взаимодействия между пользователем и компьютером должно быть удовлетворяющим желания пользователей. Чем проще общение пользователей с компьютером, тем качественнее будут исполнены желания пользователей.

Сегодня на земле живёт более семи с половиной миллиардов людей. Большая половина из них пользуются электронными устройствами. Каждый из этих людей мечтает об удобном интерфейсе. Часть из них находят удобный для себя дизайн. Всё это благодаря разработчикам интерфейсов, умеющим считывать желания людей.

## ГЛАВА 3. Эволюция интерфейса

### 3.1. Стив Джобс. Эволюция Apple

Вследствие того, что существует огромное количество различных интерфейсов, удобнее всего следует рассмотреть эволюцию одного из них. Одним из них является **Mac OS** от разработчиков фирмы Apple. «Я назову компании «Яблоко», если к пяти часам вы не предложите лучшего!» - сказал Стив Джобс своим сотрудникам. По видимому никто ничего не предложил, и теперь мы имеем отличный продукт фирмы с фруктовым названием «Apple». Более полутора миллионов пользователей мобильных устройств предпочитают продукт фирмы Apple. Основателем Apple является Стив Джобс.

Стивен Пол Джобс родился в 1955 году в городе Сан-Франциско, штат Калифорния, США.

Он закончил школу Хоумстед в Купертино. Соседом Стива был инженер Ларри Лэнг, который и познакомил Стива с персональным компьютером. С тех пор, Стив

стал интересоваться электроникой, в результате чего он познакомился со Стивом Возняком.

«Мы долго сидели с ним на тротуаре перед домом Билла и делились историями – мы рассказывали друг другу про свои розыгрыши и про разработанные нами устройства. Я чувствовал, что у нас много общего. Мне обычно тяжело объяснить людям все тонкости электротехнических устройств, которые я собирал, но Стив хватала всё на лету. Мне он сразу понравился».[\[9\]](#)

Свой первый бизнес-проект Стив Джобс осуществил со Стивом Возняком в 1971 году. Это был аппарат для розыгрыша по телефону.

С 1974 года Джобс работал в компании Atari, где «отшлифовывал» игры для симуляторов.

Компания Apple была зарегистрирована 1 апреля 1976 года. Было налажено производство компьютеров в полной комплектации. Первый заказ на 50 компьютеров был готов через месяц. Так началась история одной из лучших компаний, производящих электронные устройства.

К сожалению, 5 октября 2011 года Стив Джобс умер в своём доме в Калифорнии. 2 недели после его смерти корпоративный сайт Apple показал простую страницу с чёрно-белым портретом Джобса, при нажатии которой показывался некролог.

## 3.2. Эволюция интерфейса Mac OS

**Mac OS** (Macintosh Operating System) — семейство операционных систем, производимое компанией Apple для линейки персональных компьютеров Macintosh.

Система стала одной из первых, предоставлявших пользователю наглядный графический интерфейс вместо сложной для понимания командной строки.

Впервые Mac OS представили вместе с первым компьютером семейства Macintosh в 1984 году. Термин «Mac OS» начал использоваться лишь в середине 90-х.

В период с 1984 по 2001 годы Apple выпускала операционные системы с System 1 до Mac OS 9. Их принято считать классическими. В 2000-м году состоялся релиз Mac OS X, которая позже получила свою отдельную нумерацию.

## **System 1.0 (1984)**

Самая первая версия системы предоставляла пользователю доступ к дружественному графическому интерфейсу.

Уже тогда команде Джобса удалось сделать понятное взаимодействие с элементами в оконном режиме. По сути современные операционные системы отличаются от System 1.0 лишь графикой и дополнительными возможностями.

## **System 2.0 - 6.0 (1985-1988)**

За этот период система обрела дополнительные возможности, настройками и стандартными приложениями. Разработчики добавили полноценную иерархическую файловую систему.

Основные усилия прикладывались для адаптации ОС под новые компоненты и устройства. Появлялась поддержка слотов расширения, периферийных устройств, новых процессоров и более емких накопителей.

Лишь в System 5 была реализована полноценная модель многозадачности, позволявшая выделять системные ресурсы запущенным в фоновом режиме приложениям.

## **System 7.0 - 7.6 (1991-1996)**

Операционная система получила первый глобальный апгрейд впервые за семь лет существования.

Пользователи увидели цветной интерфейс. В большинстве системных приложений и настройках преобладали оттенки серого, но некоторые элементы стали цветными.

Многозадачность приобрела законченный вид и практически не отличалась от того, чем мы пользуемся на современных Mac.

В последующих обновлениях разработчики устраняли недоработки, добавляли поддержку новых компонентов и встраивали в систему новые приложения. Это были как собственные наработки Apple, так и приобретенное у сторонних

разработчиков ПО.

## **System 8 (1997)**

Релиз системы состоялся летом 1997 года после возвращения Джобса в Apple. Система изначально должна была продолжить старую нумерацию и выйти с индексом 7.7.

В трудные для компании времена Джобс решил поступить иначе. Apple официально перестала выпускать System 7, которая в то время была доступна для установки на компьютеры сторонних производителей.

Так в компании решили покончить с компьютерами-клонами Apple. С того времени официально операционная система компании устанавливается лишь на компьютеры собственного производства.

В Apple отчитались о 1.2 миллионах проданных копий System 8 за первые две недели после релиза и еще о 3 миллионах проданных лицензий за следующие 6 месяцев.

Сама же System 8 имела не так много изменений. Элементы интерфейса получили больше цветов и оттенков, появились темы оформления и новая панель управления.

## **Mac OS 9 (1999)**

Mac OS 9 стала последним крупным обновлением классических версий Mac OS. Релиз состоялся 23 октября 1999 года.

Разработчики логически развивали идеи, заложенные в предыдущих версиях, а ранние сборки Mac OS 9 и вовсе носили название System 8.7.

Mac OS 9 позволяла нескольким пользователям компьютера иметь свои собственные данные и системные настройки. Появилось шифрование данных по технологии Keychain, работа с удаленными сетевыми серверами и улучшенная поддержка USB протокола.

Последним обновлением классической Mac OS стала версия 9.2.2, выпущенная 5 декабря 2001 года.

## **Mac OS X 10.0 Cheetah (2001) (рис. 1)**

Первая система современной линейки Mac OS вышла в марте 2001 года. До 2012 семейства именовали «Mac OS X», затем до 2016 года апдейты получали упрощенное название «OS X», а в последние годы в компании вернулись к классическому «macOS».

Mac OS X по сути являлась преемником Mac OS 9, но, в отличие от классических систем, основывалась на Unix. В системе нашлось место многим разработкам компании NeXT конца 80-х начала 90-х годов.

Компания была основана Джобсом, а одним из его условий возвращения в Apple стало поглощение более мелкой NeXT яблочным гигантом.

В OS X 10.0 Cheetah был представлен переработанный графический интерфейс Aqua, появились новые приложения TextEdit, Preview, Mail и QuickTime.

Для работы системе требовались немалые по тем временам 128 Мб оперативной памяти и 800 Мб на накопителе.

Многие пользователи критиковали систему за отсутствие поддержки драйверов для периферийных устройств из Mac OS 9. OS X 10.0 была более требовательной и на старом железе работала менее производительно, чем предыдущие версии системы.

## **Mac OS X 10.1 Puma (2001) (рис. 2)**

Всего через полгода после Cheetah Apple выпускает крупное обновление Mac OS X 10.1 Puma.

Основной упор был сделан на повышение производительности и оптимизацию работы железа. С осени 2001 года все новые компьютеры Apple продавались с предустановленной Mac OS X 10.1, до этого многие модели комплектовались старой Mac OS 9.

## **Mac OS X 10.2 Jaguar (2002) (рис. 3)**

Разработчики сделали упор на расширение коммуникационных возможностей. В системе появляется iChat и Адресная книга.

Пользователи с ограниченными возможностями впервые получили блок специальных опций Универсальный доступ.

## **Mac OS X 10.3 Panther (2003) (рис. 4)**

В Apple довели до ума собственный браузер Safari, который стал просмотрщиком веб-страниц по умолчанию в новой версии Mac OS X.

Упор был сделан на удобную работу с несколькими приложениями и рабочими пространствами. В системе появился режим Expose.

## **Mac OS X 10.4 Tiger (2005-2007) (рис. 5)**

Выход новой версии операционной системы совпал с переходом производства компьютеров на платформу Intel. В Apple решили увеличить цикл крупных обновлений с одного года до двух.

С апреля 2005 до ноября 2007 вышло более 10 версий Mac OS X 10.4 с незначительными новшествами, улучшением совместимости и производительности.

Пользователи впервые увидели универсальный поиск Spotlight и панель Dashboard с виджетами.

## **Mac OS X 10.5 Leopard (2007-2009) (рис. 6)**

Следующие два года компания обновляла и поддерживала систему Mac OS X 10.5 Leopard. Даже параллельная разработка iPhone OS (iOS) для первого iPhone не помешала добавить ряд ключевых особенностей.



Появлялся удобный инструмент резервного копирования Time Machine, сервис Boot Camp позволил запускать на мак другие операционные системы, а сама Mac OS обзавелась поддержкой 64-битных программ.

## **Mac OS X 10.6 Snow Leopard (2009-2011) (рис.7)**

Кодовое название, незначительно отличавшееся от предыдущего, намекало, что разработчики не вносили в систему серьезных изменений, а лишь занимались доведением до ума имеющихся фишек и возможностей.

В Mac OS X 10.6 мы впервые увидели магазин приложений Mac App Store.

Тем временем в Apple окончательно отказались от архитектуры PowerPC и перешли на процессоры Intel. ОС Snow Leopard не поддерживала PowerPC.

## **Mac OS X 10.7 Lion (2011-2012) (рис. 8)**

В компании немного навели порядок с iOS и вернулись к ежегодному циклу обновлений Mac OS. Настольная система унаследовала несколько новых фишек из мобильной.

В Mac OS X 10.7 появились удобные жесты для работы с большинством стандартных приложений и системой. Дебютировал облачный сервис iCloud.

Начиная с версии Lion Mac OS перестали распространять на CD или DVD, загрузка ПО выполнялась только по сети.

## **Mac OS X 10.8 Mountain Lion (2012-2013) (рис. 9)**

Для удобства пользователей в Mac OS добавили приложения Сообщения, Напоминания, Заметки и Центр уведомлений. Программы были внешне похожи на аналогичные из iOS и дублировали их возможности на компьютере.

Добавилось несколько социально ориентированных возможностей. Из iPhone перекочевала функция Поделиться (Share), добавили интеграцию с Twitter и Facebook.

## **Mac OS X 10.9 Mavericks (2013-2014) (рис. 10)**

В Apple отошли от привычной системы названий ОС, которые в линейке Mac OS X до этого выбирали по названию видов из семейства кошачьих. Теперь операционная система именовалась в честь известного пляжа в Калифорнии.

Пользователи получили аналоги мобильных приложений Карты и iBooks, улучшенную поддержку нескольких дисплеев и возможность использовать Apple TV в качестве полноценного экрана для Mac.

Обновление впервые стало бесплатным. В компании пообещали и дальше распространять ОС на бесплатной основе.

## **Mac OS X 10.10 Yosemite (2014-2015) (рис. 11)**

Названа в честь Йосемитского национального парка.

Эта версия запомнится прежде всего серьезным редизайном. Отказ от скевоморфизма вслед за iOS способствовал сближению и более тесной интеграции настольной и мобильной ОС, которые уже и так имели общие приложения и сервисы.

Чтобы усилить сближение добавили набор возможностей под названием Handoff, но при этом ограничили поддержку таких возможностей для старых компьютеров.

## **Mac OS X 10.11 El Capitan (2015-2016) (рис. 12)**

Названа в честь самой высокой горы Эль-Капитан, расположенной в национальном парке Йосемити.

В Apple не скрывали, что данная версия системы имела множество незначительных изменений и при этом основные усилия разработчиков направили на повышение стабильности и производительности.

Улучшили большинство стандартных программ Safari, Spotlight, Mail и Заметки, добавили удобный режим Split View, но при этом обрезали функционал менее используемых программ, например, Дисковой утилиты.

## **macOS 10.12 Sierra (2016-2017) (рис. 13)**

Названа в честь горной системы Кордильер, которая находится в Калифорнии.

В компании отказались от устоявшегося наименования Mac OS X и вернулись к более классическому названию системы macOS.

В настольную систему добавили долгожданный голосовой ассистент Siri из iOS, браузер Safari получил поддержку платежного сервиса Apple Pay.

Более тесная интеграция с мобильными девайсами позволила разблокировать Mac при помощи часов Apple Watch.

## **macOS 10.13 High Sierra (2017) (рис. 14)**

Система незначительно отличается от предшественницы.

Основные изменения находятся под капотом. Система использует новую файловую систему APFS и имеет набор графических библиотек Metal 2 для разработчиков программного обеспечения.

### macOS 10.14 Mojave (2018) (рис. 15)

Новая версия macOS под кодовым названием Mojave пришла на замену High Sierra 24 сентября 2018 года.

macOS Mojave названа в честь национального заповедника Мохаве в районе между Лос-Анджелесом и Лас-Вегасе.

Новая функция Stacks позволяет навести порядок на экране, автоматически объединяя файлы одного расширения в компактные группы. Магазин приложений полностью обновил интерфейс. Выбор полезных программ стал удобнее и быстрее. Также есть тёмный режим, в котором можно выбирать типы оформления: стандартный или тёмный. Рабочий стол может меняться автоматически в зависимости от времени суток. А также улучшена безопасность OS и Safari.

Siri умеет искать сохранённые пароли, а также управлять устройствами с поддержкой HomeKit.

Документы можно редактировать в режиме «Быстрый просмотр».

Apple Почта сама предложит папку для перемещения письма и поддерживает Эмодзи.

Магазин Mac App Store полностью обновлён.

macOS 10.15 Catalina (2019) (рис. 16)

Название дано в честь крупной географической достопримечательности, расположенной в Калифорнии – острова Санта-Каталина.

Это самое крупное и значительное обновление macOS за последние годы. Самым нашумевшим новшеством является отказ от iTunes Store. Если быть точным, программа не исчезла, просто вместо одной программы, пользователи получают три: Music, Подкасты и Apple TV. Это большой плюс. Ничего никуда не надо переносить: все фильмы, музыка, подкасты уже находятся в приложениях.

Ещё одно нужное новшество – «Экранное время». Оно служит для того, чтобы контролировать количество время, проведённое перед экраном устройства.

В новой macOS Catalina не запускаются 32-битные приложения. Для любителей игр появился новый игровой сервис Apple Arcade. Пока что доступны не все игры, зато нет проблем с подключением джойстиков.

Вот так Apple медленно, но верно создает лучшую на сегодняшний день настольную операционную систему.

**Вывод:** Анализируя развитие пользовательских интерфейсов персональных компьютеров и мобильных устройств компании Apple, можно увидеть, какая огромная работа сделана её сотрудниками для того, чтобы наша жизнь была интереснее и проще. Ведь актуальность изучения современных цифровых технологий не вызывает сомнения.

## ЗАКЛЮЧЕНИЕ

Изучив тему эволюции интерфейсов, я могу сделать вывод, что средства человеко-машинного интерфейса остаются одним из ключевых систем автоматизации. Их функциональные возможности и внешний вид претерпевают ряд изменений. Какой

бы совершенной не была система, найдётся тот, кто сделает её лучше. Если рассмотреть эволюцию интерфейса Apple, можно увидеть огромный прогресс развития от первого до последнего интерфейса.

В эпоху планшетов, смартфонов и других электронных устройств, любой экран воспринимается не только как устройство вывода информации, но и как устройство ввода.

Мне интереснее всего было посмотреть эволюцию интерфейсов компании Apple за 44 года своего существования, эта компания совершила огромный прорыв в развитии информационных технологий. Началось с того, что первый интерфейс компьютера был в стиле «что видишь на экране, то и получишь». В 1984 году для повседневных задач был разработан компьютер для обывателей со сложным интерфейсом. Развитие компьютерных технологий в 90-х годах способствовало их упрощению, популярность у пользователей росла. Но вместе с тем, чтобы научиться пользоваться компьютером, надо было изучать справочные материалы. В 90-х годах интерфейс компьютеров мало изменялся и выглядел одинаково. В эволюции интерфейсов большой скачок произошёл в конце 90-х, начале 2000-х годов, когда появились сотовые телефоны. Кнопочными телефонами было сложно набирать сообщения, в результате чего появился TouchScreen. Смартфоны того времени шли в комплекте со стилусом. В наши дни мы можем управлять нашими устройствами, голосом, касанием экрана, жестом. Возможно придёт время, когда мы сможем, управлять нашими устройствами при помощи мыслей.

## **СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ**

1. Основы алгоритмизации и программирования: учеб. пособие / Т.А.

Жданова, Ю.С. Бузыкова. – Хабаровск : Изд-во Тихоокеан. гос.ун-та, 2011.

2. Программирование и основы алгоритмизации: Для инженерных

специальностей технических университетов и вузов. /А.Г. Аузяк, Ю.А.

Богомоллов, А.И. Маликов, Б.А. Старостин. Казань: Изд-во Казанского

национального исследовательского технического ун-та - КАИ, 2013.

3. Основы алгоритмизации и программирования : учебное пособие / Г.

Р. Кадырова. – Ульяновск : УлГТУ, 2014.

4. Основы алгоритмизации и программирования. Курс лекций.

## ПРИЛОЖЕНИЯ

Рисунок 1.

Интерфейс Mac OS X 10.0

[https://ru.wikipedia.org/wiki/Mac\\_OS\\_X\\_10.0](https://ru.wikipedia.org/wiki/Mac_OS_X_10.0)



Рисунок 2.

Интерфейс Mac OS X 10.1

[https://en.wikipedia.org/wiki/Mac\\_OS\\_X\\_10.1](https://en.wikipedia.org/wiki/Mac_OS_X_10.1)

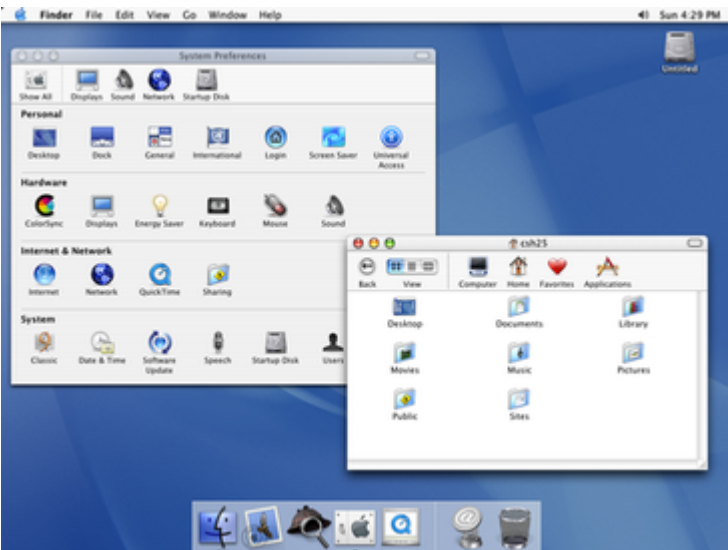


Рисунок 3.

Интерфейс Mac OS X 10.2

[https://en.wikipedia.org/wiki/Mac\\_OS\\_X\\_Jaguar](https://en.wikipedia.org/wiki/Mac_OS_X_Jaguar)

Рисунок 4.

Интерфейс Mac OS X 10.3

[https://en.wikipedia.org/wiki/Mac\\_OS\\_X\\_Panther](https://en.wikipedia.org/wiki/Mac_OS_X_Panther)



Рисунок 5.

Интерфейс Mac OS X 10.4

[https://en.wikipedia.org/wiki/Mac\\_OS\\_X\\_Tiger](https://en.wikipedia.org/wiki/Mac_OS_X_Tiger)

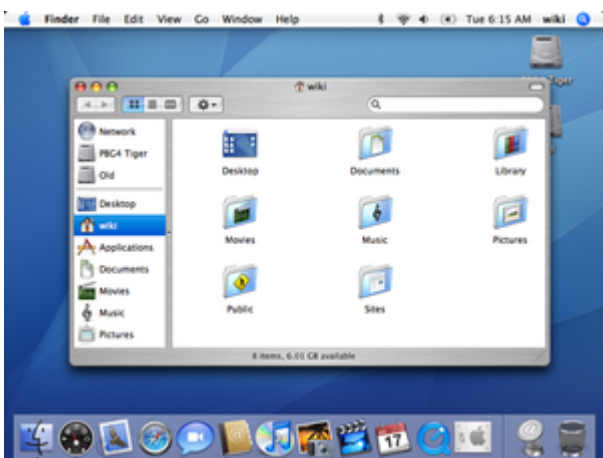


Рисунок 6.

Интерфейс Mac OS X 10.5

[https://ru.wikipedia.org/wiki/Mac\\_OS\\_X\\_Leopard](https://ru.wikipedia.org/wiki/Mac_OS_X_Leopard)



Рисунок 7.

Интерфейс Mac OS X 10.6

[https://en.wikipedia.org/wiki/Mac\\_OS\\_X\\_Snow\\_Leopard](https://en.wikipedia.org/wiki/Mac_OS_X_Snow_Leopard)

Рисунок 8.

Интерфейс Mac OS X 10.7

[https://ru.wikipedia.org/wiki/OS\\_X\\_Lion](https://ru.wikipedia.org/wiki/OS_X_Lion)



Рисунок 9.

Интерфейс Mac OS X 10.8

[https://en.wikipedia.org/wiki/OS\\_X\\_Mountain\\_Lion](https://en.wikipedia.org/wiki/OS_X_Mountain_Lion)





Рисунок 10.

Интерфейс Mac OS X 10.9

[https://ru.wikipedia.org/wiki/OS\\_X\\_Mavericks](https://ru.wikipedia.org/wiki/OS_X_Mavericks)

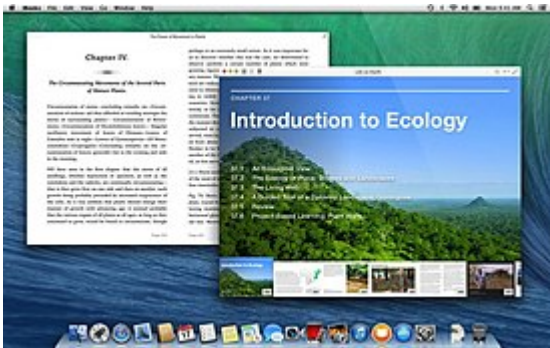


Рисунок 11.

Интерфейс Mac OS X 10.10

[https://en.wikipedia.org/wiki/OS\\_X\\_Yosemite](https://en.wikipedia.org/wiki/OS_X_Yosemite)



Рисунок 12.

Интерфейс Mac OS X 10.11

[https://en.wikipedia.org/wiki/OS\\_X\\_El\\_Capitan](https://en.wikipedia.org/wiki/OS_X_El_Capitan)



Рисунок 13.

Интерфейс Mac OS X 10.12

[https://en.wikipedia.org/wiki/MacOS\\_Sierra](https://en.wikipedia.org/wiki/MacOS_Sierra)



Рисунок 14.

Интерфейс Mac OS X 10.13

[https://en.wikipedia.org/wiki/MacOS\\_High\\_Sierra](https://en.wikipedia.org/wiki/MacOS_High_Sierra)



Рисунок 15.

## Интерфейс Mac OS X 10.14

[https://en.wikipedia.org/wiki/MacOS\\_Mojave](https://en.wikipedia.org/wiki/MacOS_Mojave)



Рисунок 16.

## Интерфейс Mac OS X 10.15

[https://en.wikipedia.org/wiki/MacOS\\_Catalina](https://en.wikipedia.org/wiki/MacOS_Catalina)



1. А.Г. Аузяк, Ю.А. Богомолов, А.И. Маликов, Б.А. Старостин / Программирование и основы алгоритмизации, 2013. - 139 с. [↑](#)
2. Научная статья на тему “Информационная технология поддержки решения математических задач” [↑](#)
3. Тельнов Ю.Ф. «Интеллектуальные информационные системы» / Ю.Ф. Тельнов. - М.: 2004. [↑](#)
4. Посевкин Н. В. Естественно-языковой интерфейс как элемент пользовательского взаимодействия // Информационные технологии.- 2015.-

№4.- С.1 [↑](#)

5. Основы алгоритмизации и программирования. Курс лекций. - 9 с. [↑](#)

6. Джеф Раскин «Интерфейс: новые направления в проектировании компьютерных систем» [↑](#)

7. С. Джобс // Fortune, 2000 [↑](#)

8. Джеф Раскин «Интерфейс: новые направления в проектировании компьютерных систем» [↑](#)

9. Возняк.- 2011.- С.83. [↑](#)