МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное автономное образовательное учреждение высшего образования

«Новосибирский национальный исследовательский государственный университет»

(Новосибирский государственный университет, НГУ)

Структурное подразделение Новосибирского государственного университета

Высший колледж информатики Университета (ВКИ НГУ)

КАФЕДРА ИНФОРМАТИКИ

**Разработка Веб-приложения NSK trip**

|  |  |
| --- | --- |
| Руководитель | Карпунькин М.  «\_\_\_»\_\_\_\_\_\_\_\_\_\_\_2020 г. |
| Студенты 4 курса  803СА2 | Мкртчян В.К. Марчак В.Е.  Михайлюк А.А. Иванов Н.Н.  «\_\_\_\_»\_\_\_\_\_\_\_\_\_\_2020 г |

Новосибирск

2021г

**Оглавление**

[**Введение** 3](#_Toc91338089)

[**Требования к Веб-приложению** 4](#_Toc91338090)

[**Анализ задачи** 5](#_Toc91338091)

[**Требование к организации процесса разработки** 7](#_Toc91338092)

[**Этапы разработки** 9](#_Toc91338093)

# **Введение**

**Наименования проекта.**

NSK trip

**Описания проекта.**

Веб приложения «NSK trip» относится к одной из самой важной частью для туризма, это путеводитель по достопремечательностям.

**Тури́зм** — временные выезды (путешествия) людей в другую страну или местность, отличную от места постоянного жительства на срок от 24 часов до 6 месяцев в течение одного календарного года или с совершением не менее одной ночевки в развлекательных, оздоровительных, спортивных, гостевых, познавательных, религиозных и иных целях без занятия деятельностью, оплачиваемой из местного источника.

**Область применения и целевая аудитория.**

Программа является продуктом сферы Туризм. «NSK trip» не имеет возрастных ограничений и может служить отличным способом времяпрепровождения.

Целевая аудитория: широкая целевая аудитория.

# **Требования к Веб-приложению**

**Требования к функциональным характеристикам**

Данный проект является Веб-приложением, вследствие чего предусматривается несколько категорий пользователей – Туризм, а так же людей кого интересует посещения достопримечательности. В процессе работы с Веб-приложением пользователь является непосредственным участник процесса и оказывает непосредственное влияние на него.

Программы должна обладать следующим функционалом:

а) графический функционал:

1) Минимальный адобтив

в) внутренний функционал:

1) просмотр мест;

2) узнать информацию о достопримечательности;

3) узнать информацию о нашем проекте;

г) интерфейс пользователя:

1) Главная страница;

2) Галерея ;

3) Карта;

4) О нас.

**Требования к составу и параметрам технических средств(для )**

Минимальные системные требования:

– ОС (операционная система): Windows XP/Vista/7/8;

– Процессор: Intel Core 2 Duo @ 3.0 Ghz / AMD Athlon 64 X2 6000+;

– Оперативная память: 1 Gb;

– Жесткий диск: 10 Gb свободно;

– Видео память: 512 Mb;

– Видео карта: nVidia GeForce 9800 / AMD Radeon HD 4870;

– Звуковая карта: Совместимая с DirectX;

– DirectX 9.0c;

– Клавиатура, Мышь.

Рекомендуемые системные требования:

- ОС (операционная система): Windows Vista/7/8;

- Процессор: Intel Core i5 @ 3.2 GHz / AMD Phenom II X4 @ 3.6 GHz;

- Оперативная память: 2 Gb;

- Жесткий диск: 10 Gb свободно;

- Видео память: 1 Gb;

- Видео карта: nVidia GeForce GTX 460 / AMD Radeon HD 5850;

- Звуковая карта: Совместимая с DirectX;

- DirectX 11;

- Клавиатура, Мышь.

# **Анализ задачи**

**Описание предметной области.**

Современное состояние автоматизации рассматриваемой предметной области

Сейчас каждый желающий может создать веб-приложение, интернет-магазин или простой информационный сайт. Для этого не нужно прикладывать много усилий, так как существует множество программ, которые делают большую часть работы самостоятельно.

Способов разработки веб-приложений много и одни из них – создание веб-приложения с нуля. Это очень трудоемкий и долгий процесс, но и он имеет свои преимущества, например, для создания следующей работы за основу можно взять уже созданный ранее шаблон. Так же создавать веб-приложения можно при помощи cms – это система, которая позволяет управлять и изменять внешний вид веб-приложения. При помощи этой системы на создание интернет-приложения уйдет в два раза меньше времени по сравнению с тем, если писать сайт самостоятельно, так как cms предлагает на выбор уже готовые шаблоны для сайта. Еще одним способом написания веб-приложения могут служить сайты-конструкторы. Их преимущество в том, что они имеют большое количество встроенных возможностей, которые упростят работу по написанию интернет-приложения

**Постановка задачи.**

Задачей данной работы является разработка Веб-приложения путеводитель для людей и туризма.

В наше время информационные технологии активно развиваются. Все больше людей начинают использовать интернет для своих нужд. Поэтому начинает появляться все больше программ, упрощающих поиск и систематизацию полученных данных, которые хранятся в информационных базах.

Базой данных (БД) называется совокупность материалов, которые систематизированы таким образом, чтобы их было легко найти и обработать с помощью ПК или другой ЭВМ (электронно-вычислительной машины). Под материалами может пониматься все, что угодно: статьи, различные документы, отчеты и т.д.

Использование баз данных для систематизации полученной информации значительно упрощается дальнейшее использование, мониторинг, хранение данных. Особенно это важно в настоящее время, когда все очень быстро меняется, например, цена на товары. В связи с эти было решено создать приложение, позволяющее вести мониторинг изменение цен в режиме реального времени с последующем сохранением полученной информации в базу данных и реализацию упрощенного доступа пользователя к накопленной информации.

**Выбор средств реализации.**

React — **это** декларативная, эффективная и гибкая JavaScript-библиотека для создания пользовательских интерфейсов. Она позволяет вам собирать сложный UI из маленьких изолированных кусочков кода, называемых «компонентами»

HTML не является языком программирования; это язык разметки, и используется, чтобы сообщать вашему браузеру, как отображать веб-страницы, которые вы посещаете. Он может быть сложным или простым, в зависимости от того, как хочет веб-дизайнер. HTML состоит из ряда **элементов**, которые вы используете, чтобы вкладывать или оборачивать различные части контента, чтобы заставить контент отображаться или действовать определённым образом. Ограждающие теги могут сделать слово или изображение ссылкой на что-то ещё, могут сделать слова курсивом, сделать шрифт больше или меньше и так далее. Например, возьмём следующую строку контента:

CSS - это язык стилей с помощью которого мы придаём стиль отображения нашего HTML контента, например придаём цвет фону (background) и шрифту, придаём контенту многоколоночный вид.

JavaScript язык программирования, который позволяет вам создать динамически обновляемый контент, управляет мультимедиа, анимирует изображения, впрочем, делает всё, что угодно. Окей, не все, что угодно, но всё равно, это удивительно, что можно достичь с помощью нескольких строк JavaScript-кода.

**Анализ данных.**

В качестве примера рассмотрим характеристики персонажа. Каждый персонаж обладает набором из 6 основных характеристик:

* сила, определяет максимально допустимый вес багажа, а также максимально возможный урон, наносимый голыми руками или оружием ближнего боя;
* ловкость, определяет координацию, скорость, рефлексы и устойчивость персонажа, точность выстрелов из огнестрельного и метательного оружия, а также вероятность уклонения от атак противников;
* телосложение, определяет общее количество здоровья персонажа, а также количество потребляемой воды и пищи;
* восприятие, определяет сообразительность, реакцию и находчивость персонажа, а также скорость его адаптации к окружающей среде;
* интеллект, определяет эрудированность и умственные способности персонажа. Чем выше интеллект, тем больше навыков может освоить персонаж.

# **Требование к организации процесса разработки**

**Модель разработки**

В «гибкой» методологии разработки после каждой итерации заказчик может наблюдать результат и понимать, удовлетворяет он его или нет. Это одно из преимуществ гибкой модели. К ее недостаткам относят то, что из-за отсутствия конкретных формулировок результатов сложно оценить трудозатраты и стоимость, требуемые на разработку. Экстремальное программирование (XP) является одним из наиболее известных применений гибкой модели на практике.

В основе такого типа — непродолжительные ежедневные встречи — «Scrum» и регулярно повторяющиеся собрания (раз в неделю, раз в две недели или раз в месяц), которые называются «Sprint». На ежедневных совещаниях участники команды обсуждают:

* отчёт о проделанной работе с момента последнего Scrum’a;
* список задач, которые сотрудник должен выполнить до следующего собрания;
* затруднения, возникшие в ходе работы.

Методология подходит для больших или нацеленных на длительный жизненный цикл проектов, постоянно адаптируемых к условиям рынка. Соответственно, в процессе реализации требования изменяются. Стоит вспомнить класс творческих людей, которым свойственно генерировать, выдавать и опробовать новые идеи еженедельно или даже ежедневно. Гибкая разработка лучше всего подходит для этого психотипа руководителей. Внутренние стартапы компании мы разрабатываем по Agile. Примером клиентских проектов является Электронная Система Медицинских Осмотров, созданная для проведения массовых медосмотров в считанные минуты. Во втором абзаце [этого отзыва](http://www.edsd.ru/public/images/otzyv-ndvision.jpg), наши американские партнеры описали очень важную вещь, принципиальную для успеха на Agile.

Когда использовать Agile?

* Когда потребности пользователей постоянно меняются в динамическом бизнесе.
* Изменения на Agile реализуются за меньшую цену из-за частых инкрементов.
* В отличие от модели водопада, в гибкой модели для старта проекта достаточно лишь небольшого планирования.

**Требования к информационной и программной совместимости**

Программа должна функционировать под управлением ОС семейства Windows следующих версий: Windows XP, Vista, 7, 8. В приложении используются библиотеки платформы.NET Framework. Также требуется установленный DirectX 9.0c или более поздней версии.

**Требование к процессу внедрения**

**Способ внедрения**

Внедрение полностью собственными силами

Преимущества :

* Меньшие финансовые затраты
* Знание бизнес-процессов
* Независимость на этапе эксплуатации

Недостатки :

* Требуются специалисты с хорошим знанием программного продукта.
* Требуются программисты
* Требуется разработка методологии управления проектом и четкое следование ей.
* Необходимость решения вопроса занятости сотрудников, выделенных (или нанятых) для реализации проекта.

**Подход к поддержке (сопровождению) и внедрению**

Прямой инжиниринг **-** процесс построения программного продукта «с нуля», исходя из проектной документации к нему.

# **Этапы разработки**

**Этапы разработки.**

Разработка web-приложения (web-разработка) — это комплекс мер и действий по планированию и созданию сайта в сети Internet в зависимости от поставленных целей и задач.

Современный Internet-сайт может решать как маркетинговые задачи — продвигать на рынке товары и услуги предприятия, конкурентные преимущества компании, увеличивать спрос на продукцию и пр. — так и представлять собой информационный ресурс, задачи которого ограничиваются информированием целевой аудитории.

Непосредственно созданию сайта предшествует подробный комплексный анализ, определяющий критерии, которым должен соответствовать будущий web-проект.

Процесс создания web-приложения, в общем случае, включает шесть основных этапов:

* определение целей и задач проекта;
* разработка структуры сайта;
* разработка дизайн-макетов;
* html-вёрстка;
* программирование и контроль качества;
* запуск и сопровождение, SEO-оптимизация.

Каждый из перечисленных этапов в достаточной степени независим, что позволяет выбирать схему работы и исполнителя для каждого из них.

Определение целей и задач проекта

На этапе проектирования формулируются бизнес-цели создаваемого проекта, определяются требования, которым он должен удовлетворять, разрабатывается общая концепция.

На данном этапе имеет смысл уточнить пожелания заказчика, а также произвести исследование целевой аудитории. Для более глубокого анализа можно запросить у заказчика соответствующие материалы: брошюры, ежегодные отчёты, образцы продукции, другие сопутствующие данные — всё, что поможет составить представление о том, кто и с какой целью будет посещать сайт, какие задачи будут выполняться на сайте. Немаловажно выяснить технические возможности будущей основной пользовательской аудитории — пропускную способность каналов связи, используемые Internet-браузеры и пр.

Для нахождения целевой аудитории целесообразно войти в роль квалифицированного пользователя на аналогичных создаваемому Internet-ресурсах. Помимо прочего это поможет выявить новые креативные концепции для того, чтобы сайт был более конкурентоспособным и «не затерялся» среди множества других.

Когда цели определены, приступают к составлению расширенного плана проекта, отражающего сколько времени, денег и иных средств понадобится для выполнения работ на каждом из последующих этапов. Такой план часто содержит информацию о бюджете проекта, график работ (с соответствующим распределением ролей между web-разработчиками), техническую документацию, а также раздел «деталей и уточнений», где оговорены конкретные аспекты возможных спорных вопросов. В этот раздел также включают предложения о готовых разработках и шаблонах.

После детального рассмотрения и утверждения плана подписывается договор о выполнении работ.

Разработка структуры сайта

Разработка структуры сайта включает всё, что касается его содержания и информационной стратегии, определяющей, как должна быть организована подача информации, чтобы будущие посетители сайта могли быстро и легко её найти. Первоочередной задачей на данном этапе является создание карты сайта, отражающей взаимосвязи типовых страниц и их наиболее значимые функциональные возможности.

Карту сайта представляют в виде чертёжа (блок-схемы), на котором каждая страница отображается отдельным прямоугольником. Связи между ними показывают схему переходов по страницам.

Также создают каркасы главной и основных типовых страниц, показывающие расположение текста и графики на странице, а также то, как пользователи будут работать с этими элементами. Каркас страницы должен предполагать возможности последующего расширения.

Более эффективную работу сайта обеспечивает соблюдение принципа: «пользователю удобнее добираться до нужной ему страницы максимум за два щелчка мыши». Поэтому обычно рекомендуется использовать не более двух уровней вложенности в пределах каждой группы элементов.

Завершив формирование облика сайта, компоновку страниц и определив размещение содержимого, переходят к следующему этапу web-разработки — к визуальному оформлению.

Разработка дизайн-макетов

Дизайн-макет — это графическое, наглядное изображение основных элементов сайта. Дизайн-макет полностью воплощает визуальную концепцию сайта. Его разработка выполняется в одной из графических программ (в подавляющем большинстве случаев — в Adobe Photoshop). В процессе разработки дизайнер руководствуется письменным соглашением (брифом) на создание дизайн-макета, который заполняется заказчиком и содержит пожелания к дизайну: тип, предпочтительные цвета, наличие тех или иных графических элементов и пр.

На этой стадии создаются все элементы web-дизайна в соответствии со стилем подачи информации и общей концепцией. Главным при дизайне сайта является умение разработать графические объекты, которые бы быстро загружались и хорошо смотрелись, независимо от используемого Internet-браузера.

Часто прибегают к использованию готовых дизайн-шаблонов, которые широко представлены в сети Internet или являются встроенными в различные графические редакторы, такие как, Microsoft FrontPage или Adobe PhotoShop. С помощью подобных шаблонов сайт создаётся за максимально сжатое время. Однако следует отметить, что у такого решения имеется ряд существенных недостатков, главный из которых — повторяемость и не уникальность дизайна. Шаблон является оболочкой с минимальным количеством интерактивных элементов и полезных модулей. Поэтому при выборе шаблона стоит обращать внимание не только на дизайн, но и на функциональность.

Важным элементом web-дизайна является графика, которую условно можно разделить на три категории:

* иллюстрированная графика — пояснительные изображения, схемы и чертежи, фотографии;
* функциональная графика — кнопки навигации, счётчики и другие элементы управления сайтом;
* декоративная графика — эстетические элементы дизайна страниц — фоновый рисунок, заголовки, выполненные в виде графических файлов и пр.

Такая классификация предполагает использование различных мультимедийных форматов. Например, для чётких, контрастных изображений с мелкими деталями и тонкими линями используются jpeg-файлы, а для красочных изображений с мягкими переходами цветов — gif-файлы.

Html-вёрстка

Html-вёрстка макета является следующим шагом после разработки дизайна сайта. Вёрстка — это преобразование созданных дизайнером графических макетов страниц в html-код, который бы отображался в Internet-браузере в точном соответствии с исходным макетом. Сложность вёрстки зависит от сложности дизайна.

Основными задачами при вёрстке являются:

* корректность отображения страниц сайта при разных разрешениях экрана;
* кросбраузерность — единообразность отображения страниц сайта в наиболее популярных браузерах — Internet Explorer, Mozilla Firefox, Opera, Chrome.

Программирование и контроль качества

Программирование — это практическая реализация проекта, интеграция наработок по отдельным направлениям. Другими словами, это процесс построения функциональных инструментов для наполнения и обработки данных. Программирование определяет насколько стабильным и защищённым будет функционирование сайта. Выбор платформы, технологий и грамотного подхода к программированию играет существенную роль. На данном этапе важно определиться с подходом к созданию Internet-реусурса: будет ли он статическим или динамическим.

Статический Internet-сайт представляет собой совокупность html-файлов, каждый из которых представляет отдельную страницу (или её часть). Такой подход используется, в основном, для размещения файловых архивов и медиа-коллекций. Статические сайты программирования, как такового, не требуют.

Страницы динамического Internet-сайт формируются сервером в ответ на запрос пользователя (передаваемый в виде URL-адреса страницы). Такие сайты требуют больше затрат на разработку и техническую поддержку, чем статические, но эти затраты окупаются за счёт снижения финансирования на редактирование материалов и развитие информационной структуры.

CMS

Как правило, современные web-проекты включают подсистему управления контентом (содержимым) — CMS (content management system), либо строятся на основе неё. Назначение системы управления контентом состоит в предоставлении возможности публиковать материалы на сайте человеку, не обладающему специальными знаниями в области создания html-документов, а также в существенном сокращении времени, затрачиваемом на ввод данных.

Такие системы делают возможным совместную работу по наполнению ресурса информацией, позволяют обновлять содержимое страниц и разделов, размещать на сайте текстовую и графическую информацию, вносить изменения, перемещать или удалять информационные блоки или целые страницы, снижают риск непреднамеренного повреждения программного кода. Управление контентом включает в себя также отслеживание актуальности опубликованной информации, её соответствия критериям SEO-оптимизации и пр.

Существует большое разнообразие системы управления контентом: от систем, создаваемых в рамках реализации конкретного проекта для решения специфических задач, до универсальных систем, таких как WordPress, Joomla, 1С-Битрикс и пр.

Инструментальные средства разработки и базы данных

В настоящее время актуальным программным функционалом среди множества сред разработки обладает высокоуровневый фреймворк Ruby on Rails. Его возможности позволяют существенно повысить продуктивность web-программирования по сравнению с такими технологиями, как PHP или Perl.

Язык Ruby, лежащий в основе фреймворка Ruby on Rails, имеет быстрый цикл разработки (редактирование – запуск – редактирование), реализован в виде интерпретатора, поддерживает нетипизированные переменные, не требующие объявления. Ruby on Rails позволяет добиться высокой надёжности и гибкости разработки проектов любого масштаба. Фреймворк включает подсистему тестирования, которая помогает существенно сэкономить время разработки и повысить качество web-проекта.

В основе большинства динамических web-проектов лежит база данных. Существует множество различных систем управления базами данных (СУБД), но в современных хостинг-центрах — организациях по размещению web-проектов на серверах — как правило, применяются СУБД MySQL и PostgreSQL. Причинами тому являются фактическая ориентация этих СУБД на хостинговые задачи, доступность на всех популярных серверных операционных системах, а также относительная простота настройки и администрирования. СУБД упрощает управление данными и сокращает время и издержки на разработку и развёртывание приложений. MySQL и PostgreSQL обеспечивают приемлемый уровень безопасности, надёжности и масштабируемости.

Системы поддержки процесса разработки

Создание web-приложения, как и любого другого программного продукта, подразумевает работу с постоянно меняющейся информацией, данными, файлами. Контроль за вносимыми в проект изменениями помогают обеспечить системы управления версиями (Version Control System — VCS), которые хранят предыдущие версии исходных файлов проекта, отслеживают производимые в файлах изменения, обеспечивают совместную командную работу над проектом и пр. К наиболее популярным на текущий момент VCS относятся: SVN, GIT, Microsoft VSS. Использование системы контроля версий поднимает общий уровень качества разработки.

Кроме того, создание любого программного продукта (в том числе и web-приложения) сопровождается новыми требованиями к продукту, предложениями по его улучшению, необходимостью исправлять ошибки и пр. Для управления перечисленными аспектами разработки и улучшения качества менеджмента используются системы отслеживания ошибок (Bug Tracking System — BTS). Главный компонент BTS представляет собой базу данных с удаленным доступом, обеспечивающую централизованный доступ ко всем необходимым файлам, спецификациям, графикам, планам, замечаниям и т.п.

Существует широкое разнообразие систем отслеживания ошибок: Basecamp, Bugzilla, Trac, MantisBT, Redmine и пр.

По завершении этапа активного программирования начинается этап тестирования корректности функционирования созданного web-приложения: проверки на наличие грамматических ошибок, пропущенных картинок, неработающих ссылок и т.д., а также проверки функционирования сайта в различных web-браузерах. Данный этап может быть автоматизирован с помощью средств автоматического тестирования таких, как IBM Rational AppScan, Empirix E-TEST Suite, XSpider, WAS и др.

Запуск и сопровождение

После исправления ошибок и презентации сайта в сети Internet, начинается новый этап работ, связанный с его сопровождением. Основная цель сопровождения — поддержание стабильности работы web-ресурса и актуальности информации. Обязательным условием квалифицированного сопровождения web-сайта является защита информации, включающая в себя антивирусную защиту и защиту баз данных на сервере от действий злоумышленников, в частности, от SQL injection.

Кроме того, необходимо своевременное обновление содержимого сайта, исправление ошибок, не выявленных или не исправленных на стадии проверки качества. Ещё одним важным моментом сопровождения является постоянный мониторинг эффективной работоспособности сайта, контроль посещаемости и учёт данных статистики.

SEO-оптимизация

Поисковая оптимизация (Search Engine Optimization) — это комплекс мероприятий, целью которых является увеличение посещаемости Internet-ресурса за счет достижения им высоких позиций в выдаче поисковых систем по заданному набору целевых запросов. SEO-оптимизация базируется на семантическом ядре — списке ключевых слов, по которым планируется продвижение сайта в поисковых системах. Ключевые слова группируются по высоко-, средне- и низкочастотным запросам. От правильности подбора данных слов напрямую зависит успех сайта.

SEO-оптимизация учитывает:

* внутренние факторы ранжирования: число, плотность и расположение ключевых слов на страницах сайта, стилистическое оформление текста, правильность формирования тегов, мета-тегов и т. д.;
* внешние факторы ранжирования: индекс цитируемости, содержание внешних ссылок, релевантность ссылающихся страниц и пр.

Современные системы web-поиска (поисковые машины) анализируют большинство элементов web-страниц: текст, заголовки, служебные html-теги, стилевые и структурные особенности содержания. Результаты работы SEO-роботов сохраняются в специальной базе данных — индексе поисковой системы, — из которой по запросу пользователя выдаются в определенном порядке. Специалисты, чьей профессиональной областью деятельности является SEO-оптимизация, постоянно изучают алгоритмы ранжирования поисковых машин.

Анализ рынка web-разработки

В настоящее время российский рынок web-разработок меняется очень быстро и, по ряду оценок, считается самым динамичным за всю историю человечества. Уже на период 2009-го года в России существовало более 2000 компаний, оказывающих подобные услуги. Среди них следует отметить Art Lebedev Group, Actis Wunderman, Айтекс, DEFA Kreativburo. Такой круг лидеров остается более-менее постоянным на протяжении нескольких лет, и только повышение спроса на редкие услуги позволяет «выбиться» наверх ранее неизвестным компаниям. Главным «рычагом» web-разработки остается рост аудитории Internet-пользователей, который стимулирует разработчиков к совершенствованию своих web-проектов и увеличению бюджетов на SEO-оптимизацию.

Сейчас в сети Internet совершаются торговые сделки, существует множество электронных валют (Яндекс.Деньги, Webmoney и другие), открыты представительства банков, где пользователи могут работать со своими счетами. Ежедневно миллионы пользователей посещают поисковые системы, электронные энциклопедии, блоги (дневники пользователей), новостные сайты. Последние годы в российском сегменте сети Internet ознаменовались бурным развитием социальных сетей, уже давно завоевавших популярность на западе.

Разработка web-приложения — это комплексный многошаговый процесс, требующий знания множества различных технологий и языков программирования, умения работать с базами данных, использовать множество инструментальных средств и программных пакетов.

По информативности Internet-сайт оставляет далеко позади все традиционные источники информации и, что не маловажно, размещенная на сайте информация доступна конечному пользователю в любое время и в любой точке мира.