**Accessibility. Как мы делаем Яндекс доступным людям с ограниченными возможностями и почему считаем это важным**

Мы внедрили поддержку accessibility на главной странице Яндекса. Сейчас слепым пользователям доступна работа с Яндекс.Браузером, [Почтой](http://mail-blog.ya.ru/16915) и частично со страницей результатов поиска, над которой работа еще продолжается. Хочу поделиться нашим опытом — возможно, он в чём-то поможет и вам или хотя бы вдохновит.  
  
Попытки обеспечения доступности интерфейсов или запуска их адаптированных версий предпринимались в Яндексе в разное время, впервые — более десяти лет назад. Однако как по организационным, так и по техническим причинам до недавнего времени систематически работать в этом направлении не получалось.  
  
  
  
Серьёзным толчком стал глобальный редизайн Яндекса. Полная переработка интерфейсов создала предпосылки для деятельности по обеспечению доступности. Не последнюю роль тут сыграл Илья [iseg](http://habrahabr.ru/users/iseg/) Сегалович, сооснователь компании Яндекс, который был сторонником внедрения accessibility и расстраивался из-за того, что этому направлению не получалось уделять должное внимание. Отдельных членов команды accessibility Яндекса привлёк именно он, причём некоторых — [прямо с Хабра](http://habrahabr.ru/company/yandex/blog/168259/#comment_5851359). Сейчас Никита [Tseikovets](http://habrahabr.ru/users/tseikovets/), на конструктивную критику которого тогда ответил Илья, один из тех, кто консультирует Яндекс по вопросам accessibility.   
  
GAAD — хороший повод рассказать, какое место в технологических процессах Яндекса занимают вопросы accessibility, как построены процессы управления проектами и разработки, а также поделиться некоторыми техническими подробностями. Мы надеемся, что наш опыт поможет и вам.

**Что такое «доступность»**

Чтобы дальше мы могли разговаривать на одном языке, разберёмся с терминами. Мы говорим то accessibility, то «доступность», поскольку термин accessibility традиционно принято переводить именно как «доступность». В русском языке этим словом обозначается много вещей — от низкой цены до лёгкого поведения. В IT под «доступностью» принято понимать availability, что не имеет отношения к accessibility. Поэтому давайте остановимся на accessibility. Сейчас этот термин в полной мере отражает деятельность по улучшению доступности продуктов и интерфейсов. Доступность — это свойство продукта, заключающееся в том, что он может быть использован самым широким кругом лиц.  
  
Целевая аудитория не ограничивается людьми с ограниченными возможностями. Зрячий человек с отключённой графикой в браузере и слепой человек во многом схожи и сталкиваются с похожими проблемами. Разница в том, что зрячие люди, отключая в браузере графику, делают осознанный выбор. У слепых такого выбора нет. Физические ограничения могут быть временными и не связанными с инвалидностью. Например, правша, сломавший правую руку, начинает испытывать проблемы с моторикой. Интерфейс, разработанный без учёта таких аспектов, может стать недоступным.  
  
С технологической точки зрения accessibility представляет собой набор практик и рекомендаций по разработке интерфейсов (сбалансированная цветовая гамма, наличие альтернативного управления с клавиатуры, использование управляющих элементов в соответствии с их стандартным назначением, добавление специальной разметки на web-страницах, реализация специальных API в приложениях и пр.).  
  
В зависимости от продукта и технологии реализации интерфейса необходимо использовать разные инструменты обеспечения доступности. Так, средства обеспечения доступности web-интерфейсов описываются [в документации W3C](http://www.w3.org/standards/webdesign/accessibility). Операционные системы имеют свои accessibility API, порой не в единственном экземпляре. Кросс-платформенные фреймворки ([Qt](https://ru.wikipedia.org/wiki/Qt_Development_Frameworks), GTK) добавляют свой API, который вызывает методы системного accessibility API. Виртуальные среды запуска (Java, Flash) также имеют свои API для взаимодействия со вспомогательными технологиями.  
  
Со стороны пользователя появляются вспомогательные технологии ([assistive technology](http://en.wikipedia.org/wiki/Assistive_technology), AT), посредством которых человек взаимодействует с продуктом и заложенным в него слоем accessibility: программы увеличения изображения, модификации цветовой гаммы, экранного доступа, программно-аппаратный комплекс управления с помощью движения головы или глаз и пр.   
  
Возникает необходимость дополнительно учитывать специфику вспомогательных технологий, которая может варьироваться от платформы к платформе, от программы к программе. Зачастую вспомогательные технологии содержат ошибки (баги), тогда надо либо их обходить, либо взаимодействовать с разработчиком технологии, донося до него суть проблемы и добиваясь её решения. При необходимости осуществляется самостоятельная разработка нужного патча.  
  
Слепые и слабовидящие пользователи пользуются компьютером и интернетом с помощью программ экранного доступа: такая программа читает вслух то, что видно на экране. У многих может возникнуть вопрос: «А зачем всё это?» По нашим наблюдениям, у компаний есть несколько мотиваций для обеспечения доступности.

* **Моральная**. Человек с ограниченными возможностями здоровья ничем не хуже остальных. Он хочет / может / имеет право пользоваться полной функциональностью сервисов и продуктов, наравне со всеми остальными.
* **Финансовая**. Пользователь с ограниченными возможностями — такой же клиент, покупающий ваши продукты или услуги. Обеспечение доступности в этом случае способствует увеличению доли рынка или выходу на новые рынки.
* **Юридическая**. Неотъемлемое право доступа к информации и услуге охраняется законами многих стран. В США и Евросоюзе принято антидискриминационное законодательство, которое обязывает разработчика учитывать интересы пользователей с ограниченными возможностями здоровья.

В случае Яндекса сработала моральная мотивация. Мы считаем, что возможность использовать наши продукты вне зависимости от особенностей человека — это важно. Экономическая мотивация теоретически может работать, но пока для нас прошло слишком мало времени, чтобы это почувствовать. В юридическом аспекте с проблемой accessibility Яндекс не сталкивался.

**Доступность в России и в мире**

Для крупных западных компаний привычно изначально закладывать в свои продукты поддержку доступности, а также иметь технологическую команду по accessibility. В России, к сожалению, о таком приходится слышать крайне редко.  
  
На Западе это во многом обусловлено юридическими стимулами. Так, компания Apple начинала свой путь к доступности с уличных демонстраций инвалидов, требующих обеспечить доступность Mac OS X, проигранных судебных дел и 250 тысяч долларов, уплаченных за недоступность iTunes. Долгую историю судебных разбирательств, публичных петиций и юридических угроз со стороны общественных организаций инвалидов США имеет Google.  
  
В США в 1998 году был принят [раздел 508](http://en.wikipedia.org/wiki/Section_508_Amendment_to_the_Rehabilitation_Act_of_1973) к закону [American Workforce Rehabilitation Act](http://en.wikipedia.org/wiki/Rehabilitation_Act_of_1973), называющийся Electronic and Information Technology Accessibility Standards и посвящённый цифровой доступности web и приложений. С 2013 года начинают входить в силу директивы нового глобального закона [Communications and Video Accessibility Act](https://www.fcc.gov/guides/21st-century-communications-and-video-accessibility-act-2010), обязывающие производителей устройств связи и воспроизведения мультимедиа обеспечивать определённый уровень доступности своей продукции. Во многом из-за этого Microsoft пришлось реализовывать минимальные функции доступности для слепых и слабовидящих в своей операционной системе Windows Phone, а Samsung — начинать внедрять доступность для слепых в своих телевизорах и телеприставках.  
  
В европейских странах также есть законы о цифровой доступности, например германский [BITV](http://de.wikipedia.org/wiki/Barrierefreie-Informationstechnik-Verordnung) или итальянский [Stanca Act](http://www.pubbliaccesso.gov.it/english/), описывающие требования к доступности сайтов. В 2014 году был принят общеевропейский Accessibility requirements suitable for public procurement of ICT products and services in Europe (EN 301 549), учитывающий доступность информационно-коммуникационных продуктов и услуг для людей с ограниченными возможностями здоровья.  
  
Многие западные компании вполне осознают и экономические выгоды от доступности своих продуктов. Например, британская сеть супермаркетов Tesco запустила проект по созданию отдельной версии своего сетевого продуктового магазина специально для людей с нарушениями зрения. После этого, по заявлениям официальных представителей компании, годовая прибыль их сети выросла на 13 миллионов фунтов стерлингов. Или, например, несколько месяцев назад сервис [Netflix заявил](http://blog.netflix.com/2015/04/netflix-begins-audio-description-for.html) о намерении продавать доступ к контенту с аудиоописанием или специальным закадровым комментарием (тифлокомментарий) для слепых. Подобный контент по закону обязаны готовить американские телеканалы, но теперь он будет представлен в агрегированном виде, что увеличит количество пользователей Netflix за счёт полностью слепых людей. Таким образом, примеров, когда доступность держится не только на социальной ответственности бизнеса, но и на обычной экономике, много.  
  
В России до этого индустрия пока не дошла, а единственный [ГОСТ Р 52872-2012](http://protect.gost.ru/document.aspx?control=7&baseC=6&page=3&month=7&year=2014&search=&id=184053) «Интернет-ресурсы. Требования доступности для инвалидов по зрению» формально является лишь рекомендательным документом.  
  
Большинство вспомогательных технологий для пользователей разрабатываются и производятся за рубежом. Серьёзных российских разработок в этой сфере пока нет. Вспомогательные технологии, широко использующиеся русскоговорящими, как правило, представляют собой локализованные продукты, что зачастую добавляет проблем, связанных со спецификой русского языка.  
  
Accessibility-технологии в России развиваются преимущественно на энтузиазме и социальной ответственности, а также в каком-то виде на формализованных требованиях к доступности, которые прописываются в технических заданиях для государственных сайтов. Очевидным образом это сказывается и на той экспертизе, которая присутствует на рынке. QA-инженеров ([Quality Assurance](http://en.wikipedia.org/wiki/Quality_assurance), контроль качества), глубоко знающих специфику accessibility, на рынке мало.

**Внедрение доступности в Яндексе**

Для определения проблемных мест и получения данных о предпочтениях пользователей Яндекс проводит опросы целевой аудитории. Эти исследования помогают определить сервисы, которыми наиболее часто пользуются люди с ограниченными возможностями, и сервисы, которыми они хотели бы пользоваться. Основываясь на собранных данных, мы формируем и корректируем нашу работу по обеспечению accessibility. Важную роль играют исследования в [usability-лаборатории](https://yandex.ru/jobs/usability), позволяющие выявить проблемы, которые пользователям трудно сформулировать в явной форме.  
  
Значительный объём работ лежит на разработчиках и QA-специалистах (инженеры по тестированию, или просто тестировщики). Бо́льшая часть работ приходится на обеспечение невизуальной доступности, то есть доступности для людей с нарушениями зрения. Это наиболее сложная задача, именно здесь применяются принципиально иные способы взаимодействия с пользователем.  
  
В общем случае процесс разработки и тестирования интерфейса с учётом доступности выглядит следующим образом.

* Первичный аудит. Интерфейс поступает на начальное тестирование к QA-инженеру, который формирует список проблем и конкретных технических рекомендаций по доработке. На этой стадии внимание уделяется базовым проблемам доступности.
* Разработка. На основе отчёта и технических рекомендаций QA-инженера разработчик проводит доработку интерфейса.
* Повторное тестирование. QA-инженер тестирует доработанный интерфейс, проверяя список изменений, список ошибок предыдущего отчёта, а также вновь проводя комплексное тестирование всего интерфейса. По итогам проверки он формирует новый отчёт и список технических рекомендаций, которые либо уточняют не до конца исправленные проблемы, либо формулируют новые задачи.
* Цикл работ. Происходит повторение шагов 2 и 3 до тех пор, пока QA-инженер не подтвердит отсутствие проблем или не признает оставшиеся проблемы приемлемыми. В ходе этого процесса могут происходить промежуточные релизы интерфейса, когда в продакшен выпускаются отдельные доработки доступности, представляющие собой более или менее законченные блоки.
* Официальный релиз. Когда решением QA-инженера по accessibility продукт признаётся доступным на достаточном уровне, происходит релиз с соответствующим статусом. Без такого заключения в релизе о доступности не заявляется.
* Количество повторений шагов 2 и 3 варьируется в зависимости от сложности работ. Например, интерфейс главной страницы Яндекса пережил четыре таких итерации, по итогам которых получил статус полностью доступного. С другой стороны, до сих пор на этапе цикла разработки находятся интерфейсы, которые пережили уже несколько итераций, но такого статуса все ещё не получили, например Поиск.

Продукты Яндекса могут выходить без статуса доступного. Однако, если об accessibility заявляется официально, это означает, что доступность была проверена не один раз и этому предшествовал серьёзный объём работ по описанной схеме. Яндекс ответственно относится к присвоению продукту статуса доступного.  
  
Насколько известно из открытых источников (и некоторой доступной внутренней информации), в крупных IT-компаниях команда accessibility, как правило, представляет собой единую группу тестировщиков и разработчиков, которые внедряются в проект и выполняют задачи по обеспечению доступности. В нашей компании применяется иной подход.  
  
В Яндексе существует большое количество обособленных друг от друга сервисов и продуктов, у каждого из которых своя команда разработки. Создание единой команды по accessibility сочтено неэффективным. Отдельная команда, конечно, может внедриться в проект и провести необходимые работы по обеспечению доступности, но остаётся непонятным, как потом основная команда проекта будет поддерживать доступность.  
  
Исходя из этих соображений в Яндексе сложилась следующая схема.

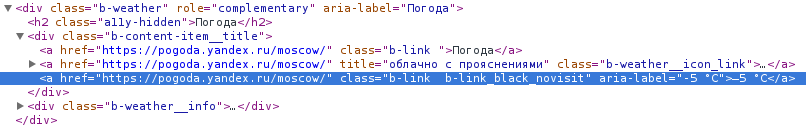
* Существует ответственный менеджер проектов, который занимается вопросами доступности. Именно он координирует работу по обеспечению доступности при взаимодействии с менеджером продукта.
* В распоряжении менеджера по доступности имеется единая команда QA-специалистов по accessibility, которые обладают знанием технологий доступности и могут как просто провести тестирование, так и предложить способ решения проблем.
* Существуют сотрудники, которые обладают глубокими знаниями в этой сфере. Это технические лидеры, которые могут помочь советом, если у разработчиков возникают проблемы. Кроме того, они занимаются разработкой внутренней технической документации по доступности.
* Наконец, в каждом проекте для accessibility выделяются специалисты из числа сотрудников, входящих в команду разработки, которые знакомятся с документацией, при необходимости консультируются с техническими лидерами и далее взаимодействуют с менеджером и QA-инженерами по доступности.

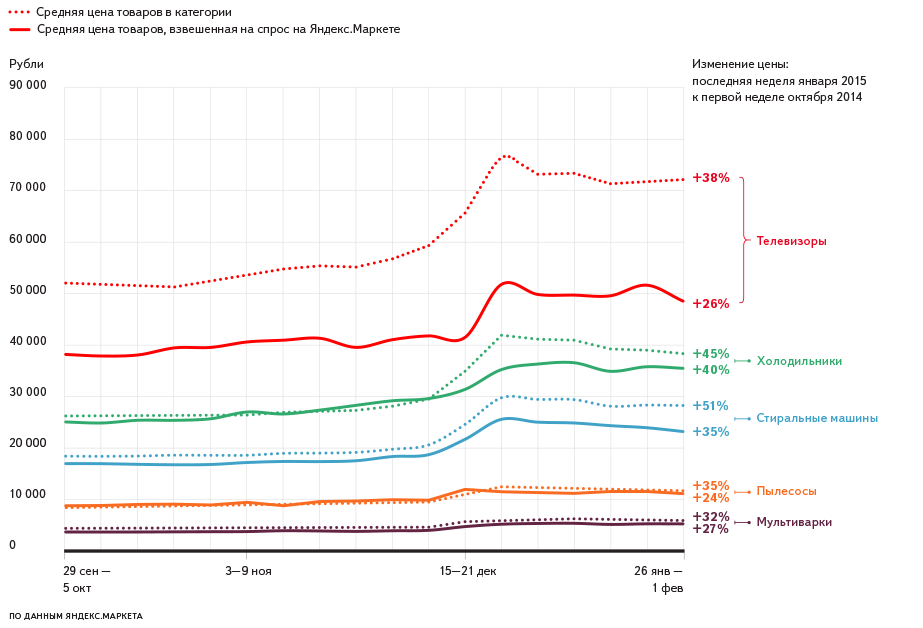
Важно помнить, что один раз обеспечить доступность интерфейса недостаточно. Обеспечение доступности — это непрерывный итеративный процесс, а сама характеристика accessibility не измеряется как она есть или её нет. Интерфейсы периодически дорабатываются, происходит добавление новых функций, редизайн и рефакторинг. Всё это может отражаться на доступности и изменять ее качество.  
  
После получения статуса доступного нельзя просто забыть об интерфейсе. Столь же важно и последующее поддержание интерфейса в форме. Для этого тестирования доступности включается в общий процесс тестирования. Кроме того, у нас есть разработки регрессионных тестов, которые в автоматизированном режиме способны отслеживать деградацию accessibility после каких-либо изменений.

**Технические аспекты внедрения доступности**

На первый взгляд может показаться, что accessibility — это дремучий лес, что в этой области очень сложно разобраться и непонятно даже, с чего начинать. Это верно лишь отчасти. В общем случае обеспечение доступности сводится к набору простых правил и приёмов работы.  
  
Важно осознать, что существуют не только мышка и touch screen и что использование клавиатуры — это нормальный способ взаимодействия с интерфейсом. Далее, следует помнить, что стандартные элементы управления интерфейса (как для web, так и для приложений) уже имеют встроенную реализацию доступности. Если использовать стандартные элементы по назначению, не придётся связываться с низкоуровневой разработкой accessibility.

**Примеры из жизни**

Впрочем, порой приходится придумывать нестандартные решения, не описанные в технических спецификациях.  
  
Например, сделайте запрос в Яндекс.Поиске с орфографической ошибкой. Над соответствующим сообщением о её исправлении вы не заметите никакого заголовка, но если откроете код страницы, то увидите, что в действительности там есть заголовок второго уровня с текстом «Дополнительная информация о запросе».  
  
Это сделано специально для пользователей программ экранного доступа, для которых заголовки являются ключевым объектом навигации. Его наличие позволяет при навигации по заголовкам сразу обратить внимание на то, что о запросе есть дополнительная информация.  
  
Поскольку у пользователей программ экранного доступа нет возможности охватить страницу взглядом, а восприятие контента идёт в линейном потоке чтения, крайне важно обеспечить информативность интерфейса, чтобы ключевые сообщения были заметны.  
  
Но, как уже говорилось, иногда приходится учитывать специфику или откровенные недоработки вспомогательных технологий, реализуя обходные пути. Например, зайдите на главную страницу Яндекса и откройте код страницы. Обратите внимание на оформление температуры в погодном блоке. Обычный текст ссылки продублирован атрибутом aria-label, предназначенным для добавления специальных меток для вспомогательных технологий. Казалось бы, зачем дублировать имеющийся текст?! За этим стоит решение достаточно серьёзной проблемы.  
  
Дело в том, что в исходной вёрстке страницы отрицательные значения температуры оформляются при помощи символа вычитания «−» (U+2212), который является мало распространённым и не опознаётся программами экранного доступа как знак минус. В итоге, запись «−5» программой читается просто как «пять», а не «минус пять». В качестве символа отрицательного числа программы чтения экрана корректно воспринимают лишь стандартный ASCII-символ дефиса «-» (U+2D). Именно поэтому здесь прописан дублирующий атрибут aria-label, в котором специально для программ чтения экрана числа отдаются со знаком дефиса, тогда как для визуального интерфейса сохранён знак вычитания.  
  


Существуют и более серьёзные проблемы, когда доступность контента реализовать сложно, — например, графики или другую изначально визуально ориентированную информацию вроде карты города.  
  
В отношении доступности SVG существуют черновики стандартов W3C. Что же касается совсем тяжёлых случаев, то тут уже речь, как правило, идёт не об обеспечении доступности существующего общего интерфейса, а о разработке параллельного способа представления информации, например, в карте города можно сделать функцию генерации текстового описания маршрута.  
  
*Сделать доступной информацию на интерактивном графике достаточно сложно. В большинстве случаев легче создать альтернативное представление.*  
  


**Невизуальный дизайн**

У web-интерфейсов с точки зрения невизуального восприятия также есть дизайн, accessibility можно реализовать по-разному, и единого пути не существует. При работе над сервисами мы придерживаемся единообразного невизуального дизайна, чтобы облегчить освоение наших интерфейсов.  
  
В невизуальном дизайне мы не оперируем понятиями цветовой гаммы или шрифта. На первый план выходят позиционирование блоков интерфейса, семантический каркас страницы и структурная вёрстка.  
  
Программа экранного доступа передаёт всё содержимое страницы в линейной форме, то есть читает её содержимое в том порядке, в котором оно размещено в HTML. Важно выделить семантические зоны, чтобы помочь пользователю понять, где какой блок интерфейса находится. Для более удобной навигации по содержимому необходимо размечать содержимое структурными тегами (заголовками, списками, таблицами и пр.).  
  
Невизуальный дизайн Яндекса строится на нескольких базовых принципах.

* Любая часть страницы должна принадлежать какой-либо семантической области.
* На странице должны выделяться базовые семантические области, описывающие шапку, поисковую форму, навигационные панели, основное содержимое и «подвал».
* Если на странице присутствуют одинаковые семантические области (области с одинаковой ролью), то они должны быть подписаны специальными метками, описывающими их назначение.
* Ключевые разделы страницы должны начинаться с заголовка (даже если заголовок невидимый).
* Перечисление связанных друг с другом общей логикой объектов должно оформляться разметкой списков.

На практике это означает, что структура страницы должна вписываться примерно в такую схему:

<body>

<div role="banner">

</div>

<div role="navigation">

</div>

<div role="main">

</div>

<div role="contentinfo">

</div>

</body>

Содержимое не должно выходить за пределы обозначенных div, чтобы не оказаться вне семантической области.  
  
Вот более полный пример с иерархией областей, дополнительными метками и структурной разметкой:

<body>

<div role="banner">

<div role="search">

</div>

</div>

<div role="navigation" aria-label="Другие сервисы">

<ul>

<li>Сервис 1</li>

<li>Сервис 2</li>

</ul>

</div>

<div role="navigation" aria-label="Разделы">

<ul>

<li>Раздел 1</li>

<li>Раздел 2</li>

</ul>

</div>

<div role="main">

<div role="complementary" aria-label="Информация 1">

<h2>Один блок информации</h2>

</div>

<div role="complementary" aria-label="Информация 2">

<h2>Другой блок информации</h2>

</div>

<div role="complementary" aria-label="Реклама">

</div>

</div>

<div role="contentinfo">

</div>

</body>

Слепой пользователь имеет возможность перемещаться по структурным элементам, таким как заголовки или списки, а также понимает, где какая область страницы находится, и может быстро по ним перемещаться, при этом не рискуя, что потеряет какой-то контент, так как весь контент находится внутри целевых семантических областей.  
  
В Яндексе занимаются доступностью не только web-интерфейсов, но и приложений, то есть application accessibility. На сегодняшний день в продакшене это Яндекс.Браузер под Windows.

**Доступность приложений**

Обеспечение доступности приложений существенно отличается от обеспечения доступности web-интерфейсов. Дело в том, что в случае web мы работаем с универсальными кросс-платформенными технологиями. Да, пользовательские вспомогательные технологии зачастую имеют определённую специфику и порой требуют её учёта, но всё-таки основные принципы реализации едины, то есть не зависят от операционной системы и браузера. Ну а при обеспечении доступности приложения технологии accessibility начинают варьироваться в зависимости от операционной системы, а специфика пользовательских вспомогательных технологий оказывается более существенна.  
  
Например, в процессе работы над доступностью Яндекс.Браузера под Windows нам пришлось решать сразу две задачи.

* Во-первых, используя стандартные системные accessibility API, мы обеспечили доступность инсталлятора и меню приложения.
* Во-вторых, столкнувшись с тем, что область просмотра страницы была недоступна для одной из наиболее популярных программ экранного доступа, мы самостоятельно разработали необходимый патч для этого экранного чтеца и, связавшись с российским дистрибьютором этого программного обеспечения, договорились о включении патча в последующие сборки.

Проблема в том, что доступность Windows-версии Яндекс.Браузера абсолютно никак не поможет при обеспечении его доступности под OS X или iOS. Это отдельный проект, который надо будет реализовывать и поддерживать параллельно, так как кросс-платформенных accessibility API в чистом виде не существует.

**Что доступно в Яндексе**

Разработка доступности в Яндексе в первую очередь ориентируется на пользователей с нарушениями зрения, которым приходится иметь дело с программами экранного доступа (преимущественно, [JAWS](https://en.wikipedia.org/wiki/JAWS_%28screen_reader%29), [VoiceOver](https://en.wikipedia.org/wiki/VoiceOver) и [NVDA](http://en.wikipedia.org/wiki/NonVisual_Desktop_Access)). Однако не следует забывать, что это не единственные пользователи с ограниченными возможностями здоровья.  
  
Так как на данный момент не существует технологий для отслеживания слепых пользователей, в Яндексе используются практики анкетирования слепых и слабовидящих пользователей и проведения usability-исследований. Опрашивая пользователей, мы узнаём ценную информацию о наших продуктах.  
  
Традиционно информация о потребностях пользователей оказывает большое влияние на приоритеты в разработке. Так, нам важно знать, какие сервисы уже применяют наши пользователи с нарушениями зрения и какими хотели бы пользоваться. К сожалению, мы не можем работать над доступностью всех наших сервисов одновременно, поэтому в первую очередь занимаемся основными:

* Главная страница в доменах .ru, .ua, .kz, .by,
* Поиск,
* Почта,
* Яндекс.Браузер.

Некоторые сервисы уже хорошо доступны, по другим работа ведется.

**Послесловие**

Обеспечение доступности web-сервисов и приложений для самой широкой аудитории —важная и благородная цель. К этой цели нужно идти последовательно, шаг за шагом. Главное — не сходить с пути, не бросать дело на полдороге и не делать вид, что людей с ограниченными возможностями здоровья не существует. Тогда в один прекрасный день вы почувствуете, что делаете мир лучше.

Источник: http://habrahabr.ru/company/yandex/blog/258477/