

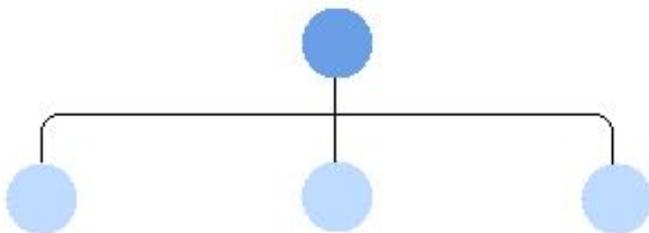
***Основные принципы
к построению
пользовательского
интерфейса***

Пользовательский интерфейс – система правил, принципов, средств, регламентирующая и обеспечивающая взаимодействие программного продукта с пользователем.

Дизайн пользовательского интерфейса – проектирование программных продуктов, соответствующих практическим, физическим и эмоциональным потребностям широкого круга пользователей.

Принцип простоты и адекватности

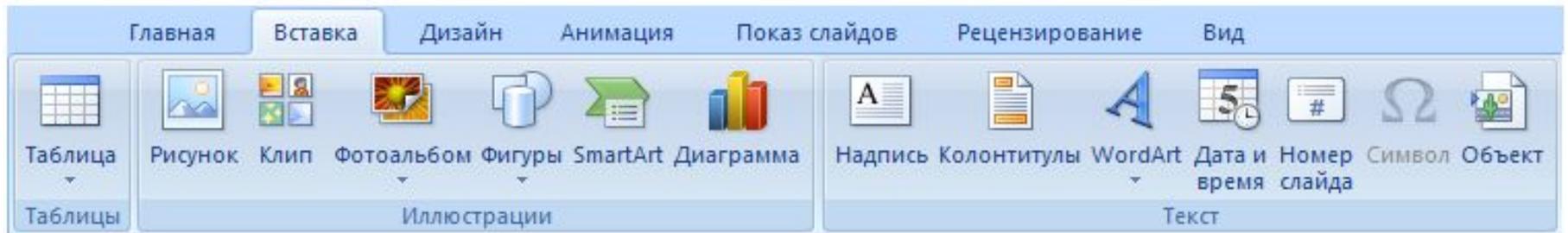
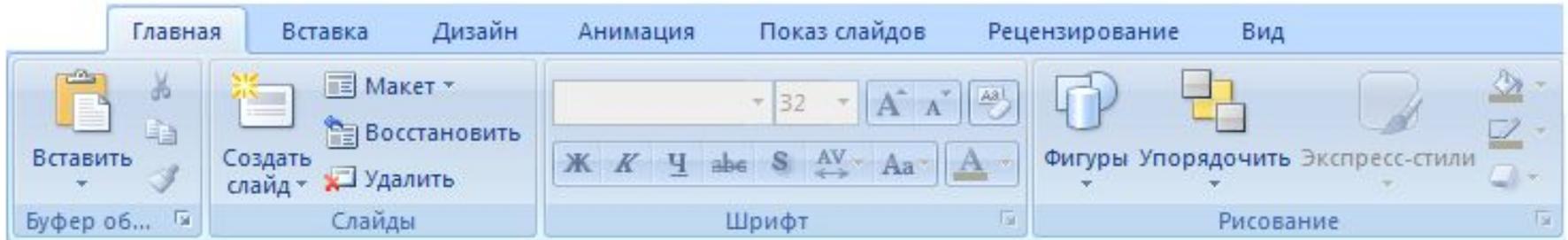
К пользовательскому интерфейсу предъявляются два противоречивых требования: **простота и адекватность предметной области**. Основная проблема при разработке пользовательского интерфейса состоит в нахождении компромисса между его простотой и необходимостью учета многочисленных особенностей. Попытка построить единый универсальный пользовательский интерфейс с большим количеством нагрузочных параметров, несомненно, обречена на неудачу. Для достижения компромисса между простотой, и в тоже время его адекватности, необходимо однозначно определить какие подсистемы, элементы и взаимосвязи между ними войдут в пользовательский интерфейс а какие будут носить вспомогательный по



Продуманный и не продуманный пользовательский интерфейс

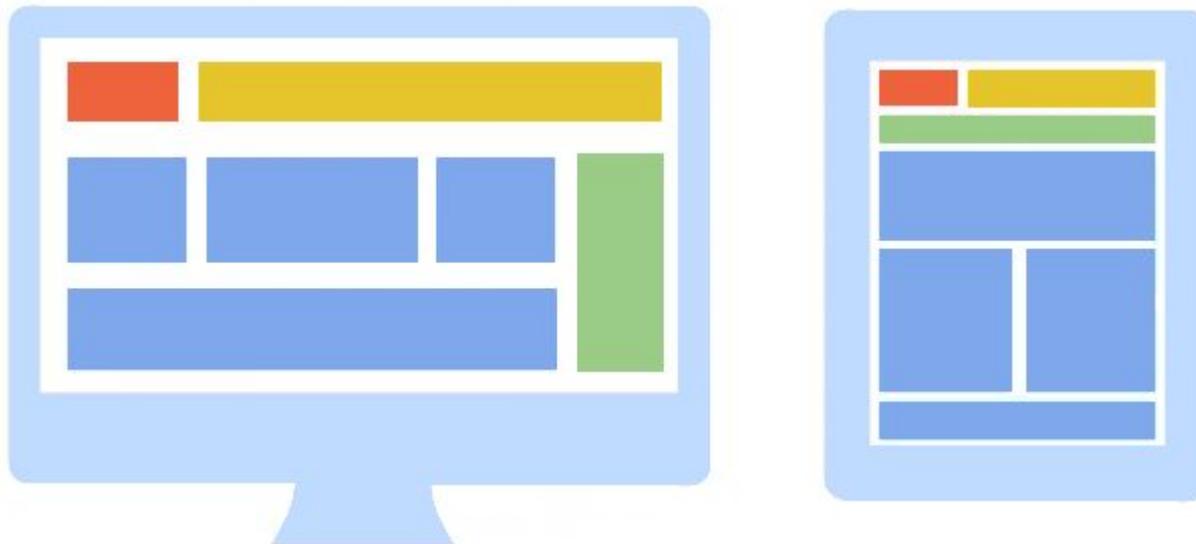
Принцип стилистической целостности

Стилистическая целостность пользовательского интерфейса достигается единообразием оформления, под которым подразумевается **ограничения в выборе гарнитур и размеров шрифта**, а также равномерное распределение свободного пространства.



Принцип выравнивания

Ни один элемент пользовательского интерфейса не должен располагаться случайным образом, т.е. **нельзя располагать элементы или подсистемы где-либо** только потому, что там присутствует свободное пространство.



Принцип выравнивания помогает создать визуально уравновешенный дизайн пользовательского интерфейса

Принцип уменьшения визуального шума

Визуальный шум является одним из основных препятствий для получения комплексного информационного представления о системе.

Можно выделить два типа визуального шума.

1) Перегруженность. При работе с пользовательскими интерфейсами, имеющими значительные размеры, большое количество всевозможных подсистем, элементов и бесчисленное множество различных цветов и яркостей, зрительная система пользователя испытывает повышенную нагрузку.

2) Фоновый шум. **В качестве фона пользовательского интерфейса рекомендуется применять малонасыщенные и нейтральные цвета** средней частоты спектра. Не допускается использование в большом количестве цветов, которые затрудняют восприятие.

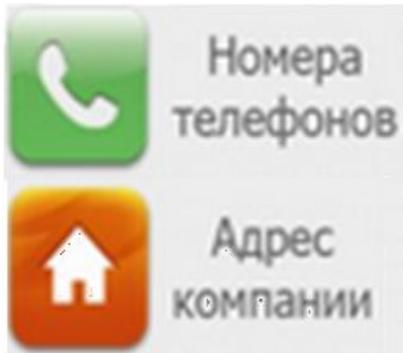
Принцип использования привычных ассоциаций и стереотипов

Сущность данного принципа состоит в применении для пользовательских интерфейсов таких условных обозначений элементов, которые ассоциируются с общепринятыми буквенными обозначениями данных элементов. Желательно использовать, если это возможно, вместо абстрактных знаков и символов, графические образы, которые ассоциируются с элементами. ***Если изображение или его часть знакомо пользователю, то он сможет идентифицировать его даже по его фрагменту.*** Любые образы должны давать однозначное представление о функции. Неграмотное использование графики может дезориентировать пользователя в программе, исказить восприятие.



Принцип избыточности

Для улучшения восприятия структуры системы иногда бывает **необходимо параллельное представление наиболее важных компонентов** или взаимосвязей.



Названия элементов продублированы графическими образами



Повтор панели навигации внизу экрана более простом оформлении

Принцип повтора

Некоторые компоненты стилового оформления должны повторяться, что делает пользовательский интерфейс более цельным, согласованным и упорядоченным.

Повторяться могут маркеры списков, шрифтовое оформление (кегель, жирность), цветовые решения, пространственные взаимосвязи, толщина соединительных линий, распределение свободного пространства и т.п.

Положительная роль повтора очень сильна в простых приложениях, а в сложных без него не обойтись.

При отсутствии повторяющихся элементов, каждая страница лишится связности и стилового единства. Но необходимо избегать избыточного повтора, который будет подавлять смысловое содержание.

Принцип визуальной иерархии

Элементы, являющиеся частями друг друга, представляются в виде вложений. Необходимо организовать внешний вид элементов таким образом, чтобы **пользовательский интерфейс ясно и четко отображал отношения между элементами** – какие элементы связаны между собой, какие являются частями других элементов (подсистем)

Мультимедийные диски
Компакт-диски для детей
Компакт-диски с учебными курсами
Развлекательные компакт-диски
DVD-диски
Образовательные
Раннее обучение
Обучение чтению, письму, словесности
Наука
Математика
Для учителей

Книги
Рабочие журналы для учителей
Видео
Оборудование и комплектующие
Кабели
Устройства ввода
Накопители
Память
Модемы
Принтеры и расходные материалы
Фильмы и музыка

Для учителей
Книги
Рабочие журналы для учителей
Видеокурсы

Оборудование и комплектующие
Кабели
Устройства ввода
Накопители
Память
Диски
Принтеры и расходные материалы
Фильмы и музыка

Мультимедийные диски
Компакт-диски для детей
Образовательные компакт-диски
Развлекательные компакт-диски

Образовательные
Раннее обучение
Обучение чтению, письму, словесности
Наука
Математика

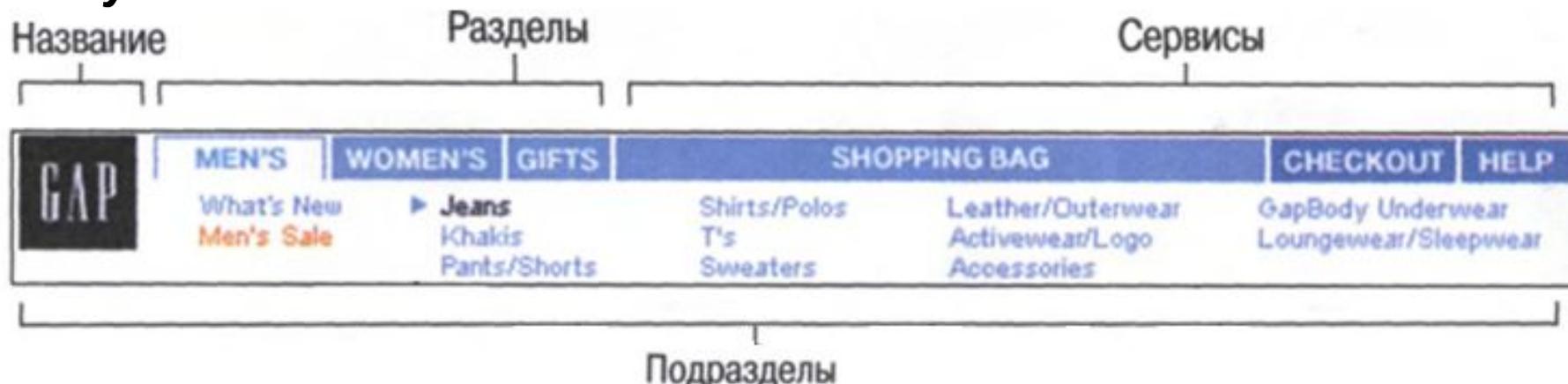
Очевидно, что данный список элементов необходимо сделать понятнее.

Тот же список, визуально разбитый на группы.

Принцип удобства использования

Важным моментом при создании интерфейсов является «usability» (usability — удобство и простота использования). При организации пользовательского интерфейса необходимо руководствоваться не только аспектами красоты, но и практическим удобством расположения управляющих элементов. Пользователь не должен долго искать необходимую ему функцию.

Управляющие элементы пользовательского интерфейса должны располагаться там, где пользователь ожидает их увидеть.



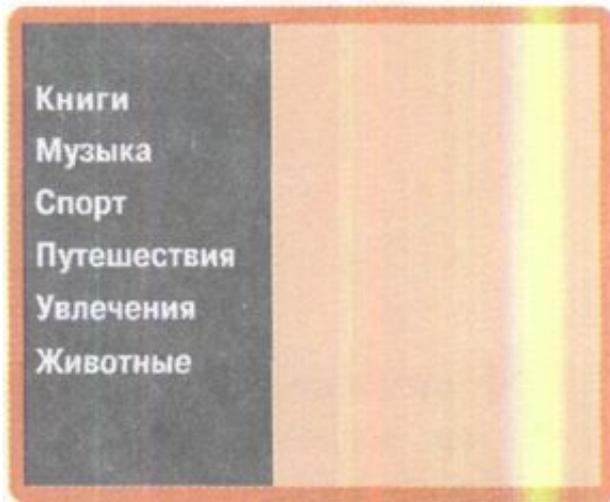
Принцип визуальной иерархии

Каждый элемент пользовательского интерфейса необходимо взвешивать по его способности помочь пользователю воспринять структуру приложения. ***Пользовательский интерфейс не должен содержать лишних, затемняющих его элементов.*** Применять элементы, обозначающие несущественные особенности приложения, не следует. Необходимо определить относительную важность каждого из элементов модели по отношению к поставленной системной цели.

Необходимо обособить элементы пользовательского интерфейса, обладающие относительной автономностью. Эти обособленные участки должны быть четко отграничены от других и согласно принципу структурности должны иметь завершенную, легко запоминающуюся и отличающуюся структуру.

Принцип визуальной иерархии

Равнозначные элементы необходимо обозначать одинаковыми графическими объектами, сгруппировывать под одним заголовком, применять к ним одинаковый стиль оформления или размещать их в единую четко определенную область пользовательского интерфейса.



Элементы, логически связанные между собой, должны быть связаны и визуально.



Чем более важен элемент, тем более он заметен.

Принцип визуальной иерархии

Значение графического элемента должно определяться его окружением. ***Желательно выделять размером, формой или цветом элементы, наиболее существенные для принятия решений.*** Например, важные элементы отображаются несколько крупнее, выделяются цветом, отделяются свободным пространством, размещаются ближе к верхней части или центру или используются различные комбинации данных способов.

Кроме того, желательно снизить нагрузки (особенно когнитивную и визуальную) для облегчения восприятия и использования пользовательского интерфейса. Но иногда необходимо увеличить ту или иную нагрузку. Например, ***чтобы акцентировать внимание, можно расширить интерфейс дополнительной визуальной информацией*** и тем самым повысить его нагруженность (основные факторы визуального веса: цвет, контраст, яркость, размер, плотность и т.п.).

Принцип визуальной иерархии

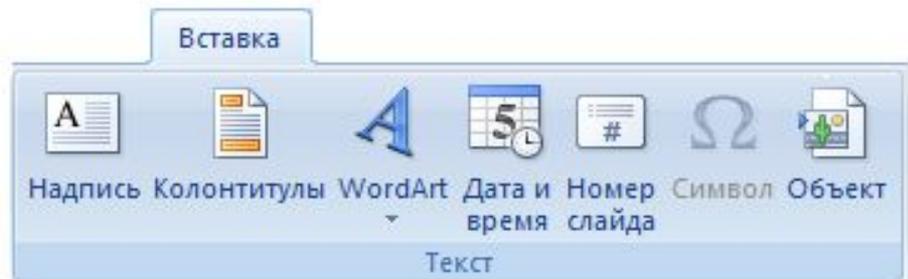
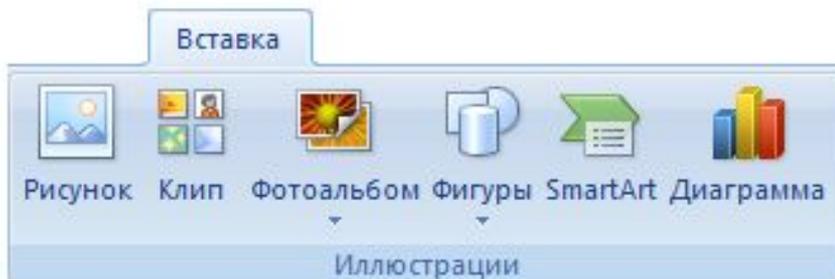
Логически связанные между собой элементы должны быть связаны визуально. Связанные элементы обычно объединяют в группы, тогда они воспринимаются как единое целое, а не как скопление артефактов. ***Сбалансированная иерархия помогает не только задать порядок восприятия пользовательского интерфейса***, но также помогает объединять отдельные элементы в единое целое. Ни один элемент приложения не должен занимать случайную область пространства, тогда они образуют один визуальный компонент. Это способствует организации информации, уменьшению визуального шума и повышает восприятие приложения в целом. Но если структурных элементов большое количество, то структурировать необходимо лишь те элементы, между которыми можно установить четкую связь. Таким образом, ***в компонентном составе должны присутствовать ясно выделенные подсистемы элементов***, которые желательно ассоциировать с некоторым заголовком.

Принцип визуальной иерархии

Американский ученый-психолог Джордж Миллер обнаружил закономерность, суть которой состоит в том, что **кратковременная человеческая память может запомнить и повторить только 7 ± 2 элемента.**

Соответственно группировать элементы системы желательно с учетом этого правила – то есть не более семи в группе, в крайнем случае – девяти. **Такие небольшие группы объектов наиболее хорошо воспринимаются пользователями без использования каких-либо специальных приемов.** При этом группировка,

используемой



Принцип перевернутой пирамиды

Смысл этого принципа состоит в том, чтобы вынести самую важную информацию и элементы управления на первый план и сделать их легкодоступными пользователю.

Только небольшая часть посетителей просматривают информацию за пределами окна браузера, в котором они первоначально увидели содержимое. Исходя из этого, желательно размещать содержимое в виде перевернутой пирамиды:

1. Начать с того, что быстро даст пользователю представление о сути содержимого страницы. В данном случае пользователи могут завершить сеанс взаимодействия в любой момент и будут уверены, что узнали главное.

2. Содержимое, которое менее важно или не существенно для понимания располагается внизу.

Принцип перевернутой пирамиды



Принцип эмоционального воздействия

Цвет вызывает эмоции. Например, красный цвет – волнующий, желтый – веселый и т.п. Разрабатывая пользовательский интерфейс в ярких цветах, необходимо подумать об эмоциях, которые они будут вызывать у пользователя.

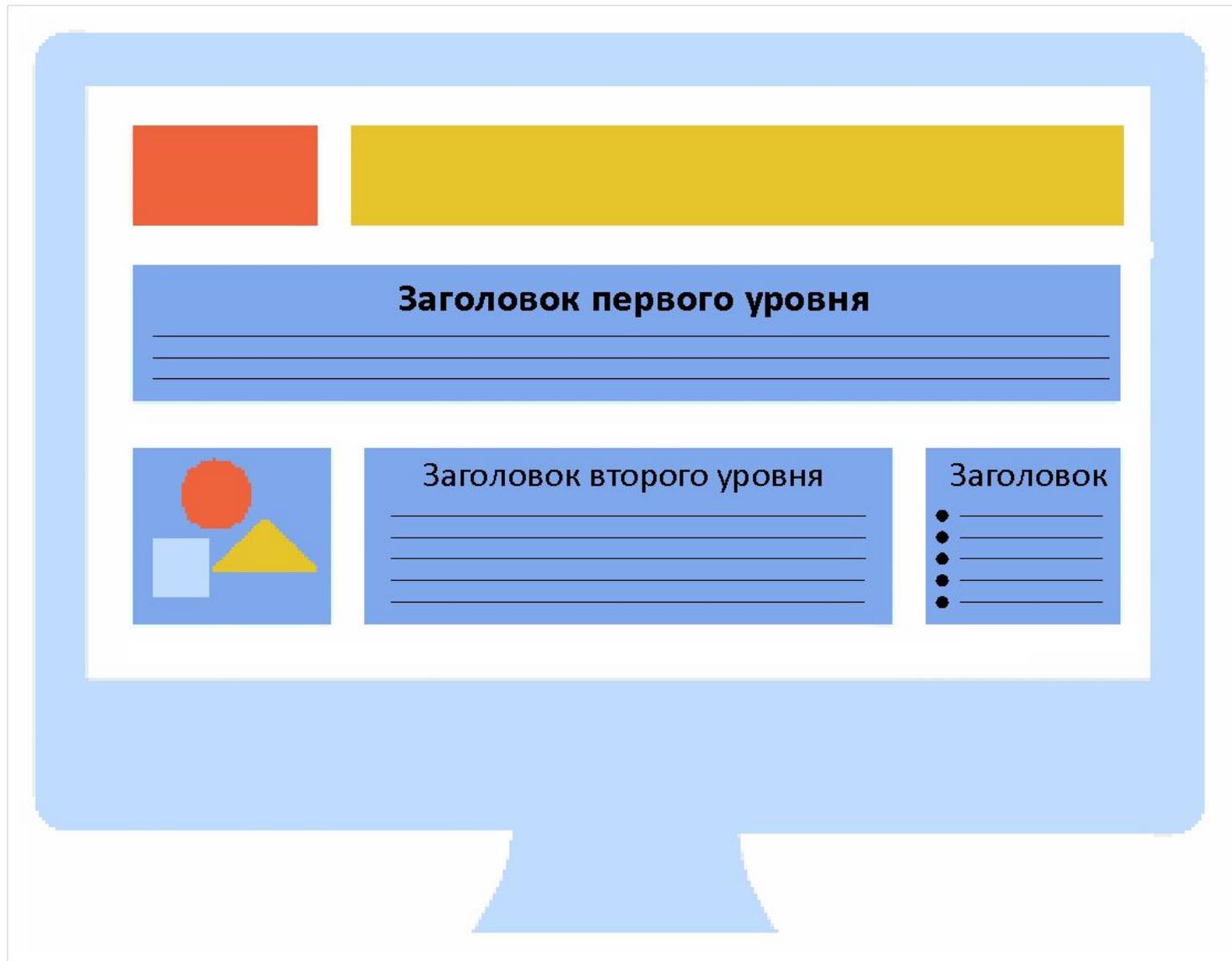
Цвет является элементом, который так же важен, как навигация, изображения или содержимое. Интерфейс с грамотно подобранной цветовой гаммой создают ощущение, которое не возможно получить от интерфейса, где цвета не используются в качестве элемента дизайна или просто плохо подобраны.

Принцип сканируемости

В Сети пользователи не читают текст так, как с печатного листа. Вместо этого они сканируют информацию, ища ключевые слова, предложения и изображения, интересные им.

Заголовок является хорошим способом повысить сканируемость текстовых блоков. Заголовки превращают большие фрагменты текста в меньшие, что позволяет посетителям быстро решить, читали ли фрагмент. Заголовок должен содержать максимум три строки, иначе он будет хуже восприниматься при чтении.

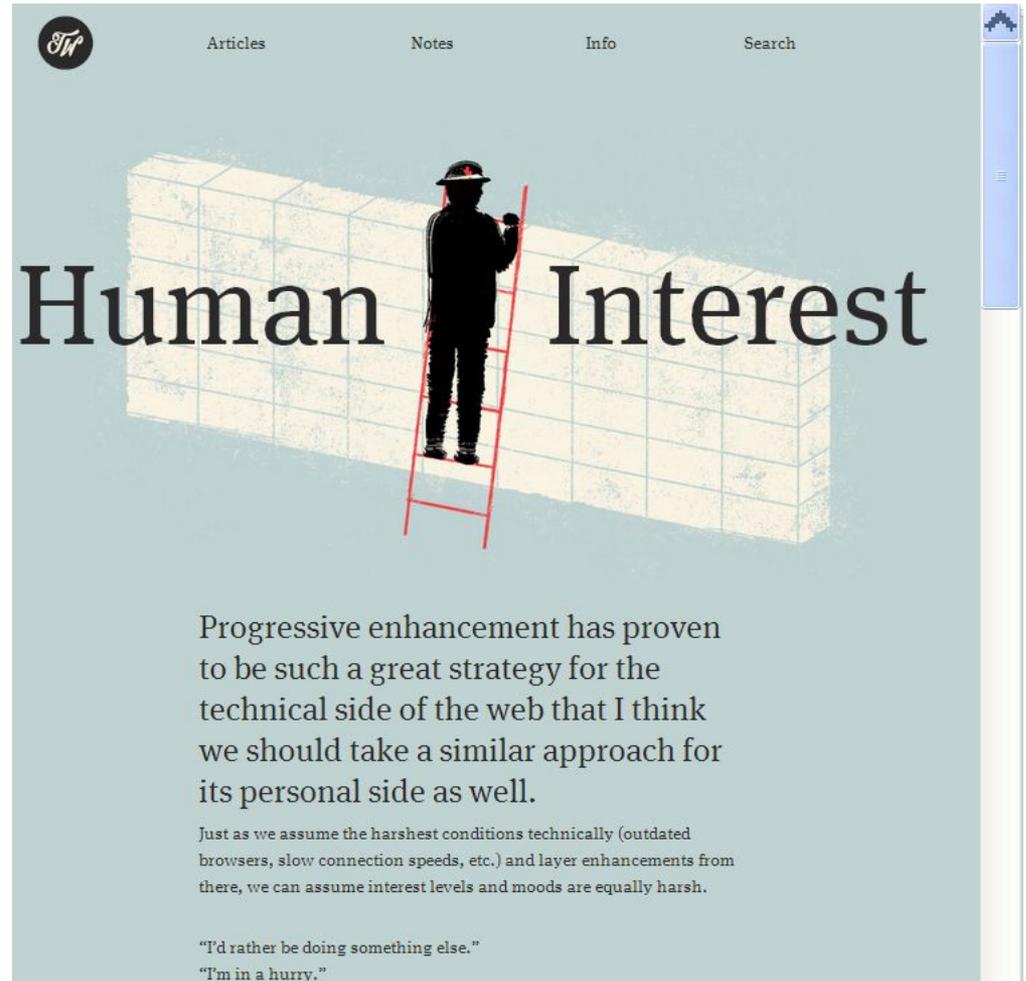
Принцип сканируемости



Принцип сканируемости

Заголовки могут несколько удлинить страницу, но для пользователей не составит труда прокрутить ее вниз, если им интересно ее содержимое.

Если контентная часть отредактирована в соответствии с правилом перевернутой пирамиды, то пользователи быстро определяют, нужен ли им данный контент.



Articles Notes Info Search

Human Interest

Progressive enhancement has proven to be such a great strategy for the technical side of the web that I think we should take a similar approach for its personal side as well.

Just as we assume the harshest conditions technically (outdated browsers, slow connection speeds, etc.) and layer enhancements from there, we can assume interest levels and moods are equally harsh.

"I'd rather be doing something else."
"I'm in a hurry."

Принцип контроля

Пользователи чувствуют себя уверенно, когда владеют ситуацией. Плохо спроектированные пользовательские интерфейсы вынуждают пользователя выполнять незапланированные действия, которые мешают достижению конечной цели.

Необходимо сообщать пользователю о текущем состоянии, в которой он (или программное приложение) находится в данный момент, периодически отображая системный статус, объясняя что произойдет, если он сделает то или иное действие.

