**Практическая работа № 1.**

**Человеко-компьютерное взаимодействие. Человеко-компьютерный интерфейс**

Согласно своему варианту выполнить задание практической работы:

**Задание 1.** Определить тематику для разработки пользовательского интерфейса.

|  |  |
| --- | --- |
|  | Аэропорт (Ж/д станция). Пассажирское расписание и перевозки |

**Задание 2.** Подготовить презентацию (см. вариант)

|  |  |
| --- | --- |
| *Вариант* ***3****.* | Определить основные поколения интерфейсов. Распечатать отчет и защитить теоретическую часть. |

**Ссылка для презентации:** <https://docs.google.com/presentation/d/16w9LFaN5kwRX4stABl1j3UITehbHnoSj/edit?usp=sharing&ouid=101549415473444107374&rtpof=true&sd=true>

Ссылка на весь проект(figma):

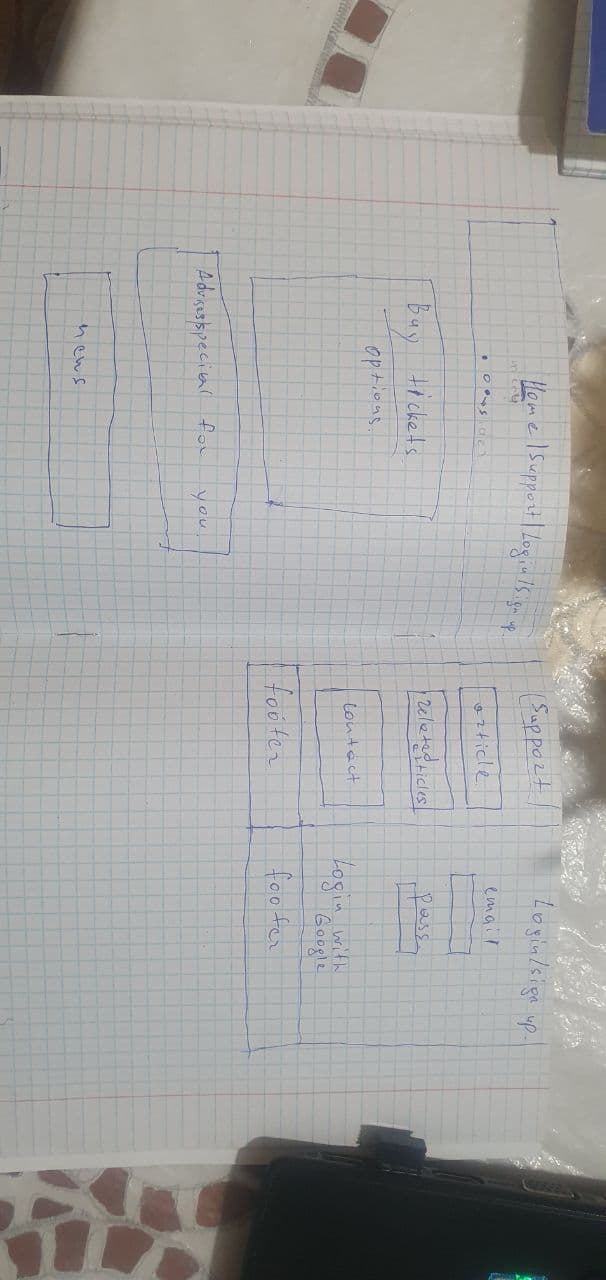
<https://www.figma.com/file/3sOvP6uFeA1Qjrf05Yrafn/TicketAir?node-id=1%3A2>

**Практическая работа № 2**

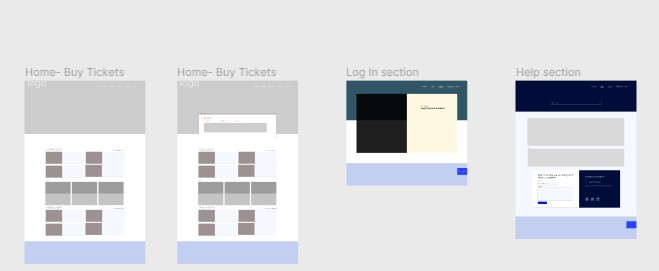
**Дизайн дисплея. Требования к построению прототипа. Этапы построения прототипа: создание версий.**

**Задание.** Согласносвоей темы (см. практическая работа 1) создать 2 разные версии прототипа интерфейса.

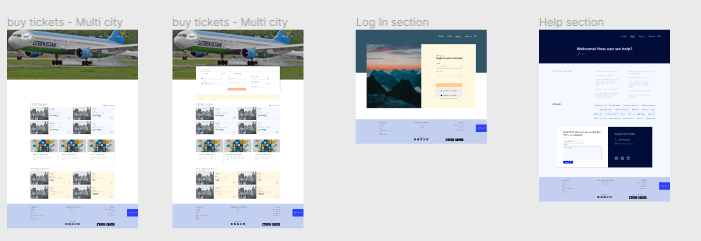
1. Концептуальный (очень интересно но ничего не понятно(я составил мысленно и с эмоциями(^\_\_^)))
2. Бумажный хотя я не художник, но посторался))



1. Псевдореальный



1. Реальный



**Практическая работа № 3.**

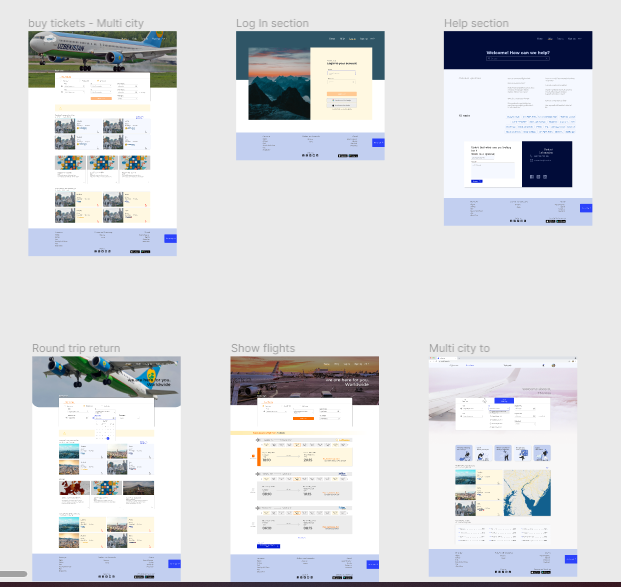
**Принципы и правила использования компьютерной техники. Построение прототипа интерфейса.**

**Цель работы:** получение навыков проектирования прототипов прикладного программного обеспечения автоматизированных систем.

**Задачи работы**: освоение принципов применения программного продукта EaSynth ForeUI для проектирования прикладного программного обеспечения.

**Задание.** Согласносвоей темы (см. практическая работа 1) построить прототип интерфейса.

**Часть готового MVP:**



Полный проект: <https://www.figma.com/file/3sOvP6uFeA1Qjrf05Yrafn/TicketAir>

**Практическая работа № 4.**

**Аудит удобства, тест. Закон Хика. Использование закона Хика на практике.**

**Общие сведения**

Чем больше вариантов, тем больше времени уходит на выбор между ними. Эта аксиома, хорошо известная разработчикам интерфейсов как закон Хика, должна учитываться при написании программного обеспечения. Однако при работе со многими программными продуктами возникает ощущение, что их цель – отнять как можно больше времени у пользователя, или что закон Хика при их создании был забыт.

**Задание**

1. Согласно описанным методам произвести оценку быстродействия и эффективности разработанного интерфейса приложения: определить сколько времени затратит пользователь для достижения указанной в ТЗ функциональности, определить сколько лишней информации содержится в интерфейсе.

**Проведём опыт на основе закона Хика.**

**Пример 1.**

В этом примере показано главный экран приложения “TicketAir”.Мы на примере этого приложения проводим опыт. В рисунке отображено главный экран приложения, где ползователь выбирает нужную себе услугу и проходит в следующий экран, где отображается конкретные подуслуги(если так можно сказать). Поьзователь при этом выбирает услугу а потом выбирает для себя подходящего маршрута.

И так начинаем:

Формулировка закона Хика для выбора из n равновероятных вариантов:

**t (время выбора) (мс) = a+b\*log***2***(n+1)**

t — это общее время реакции,

a и b — константы, которые описывают индивидуальные особенности восприятия, такие как задержка перед выполнением задания и индивидуальный коэффициент скорости принятия решения,

n — количество равнозначных альтернативных вариантов, из которых нужно выбрать.,

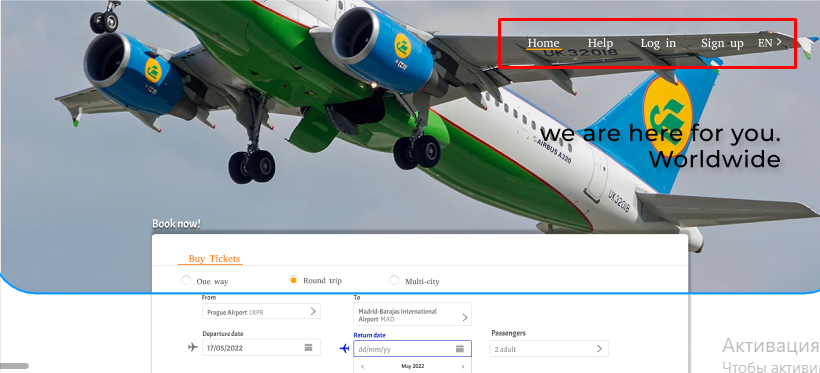
В примерах a=45, b=155мс. (155мс = 0,155с)

**Опыт 1**. Прии входе в главнную станицу в меню есть 5 объектов для выбора.

Для выбора одного элемента из восьми

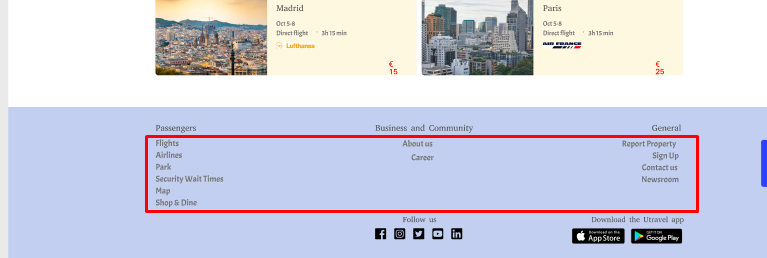
|  |
| --- |
| 445,6691876 |

=

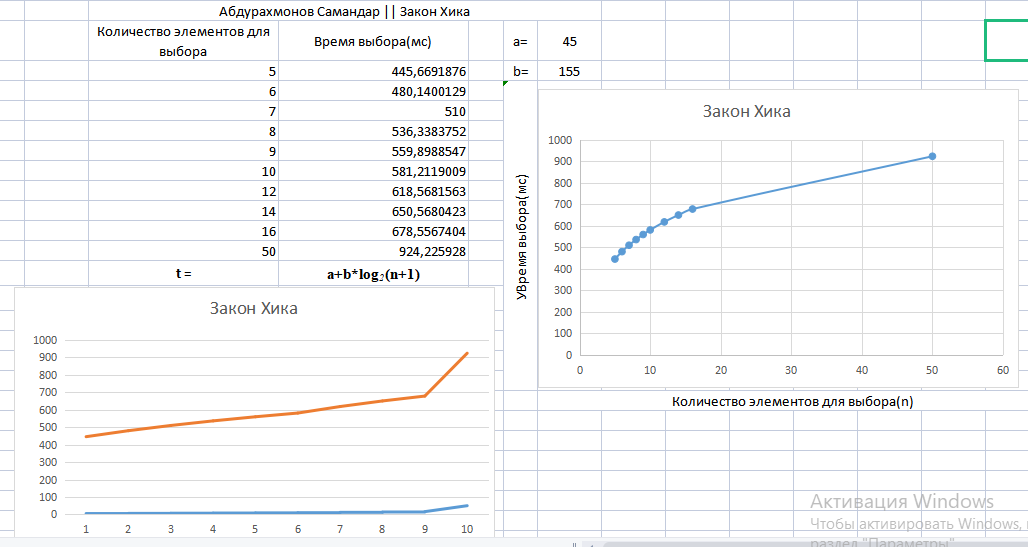


**Опыт 2**. Для выбора из 12 элементов :

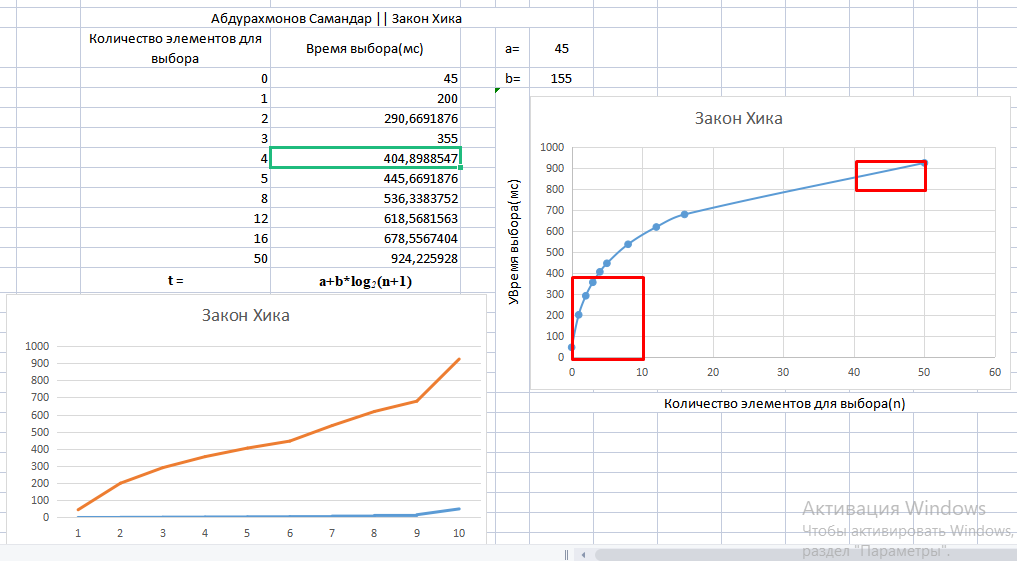
|  |
| --- |
| = 618,5681563 |

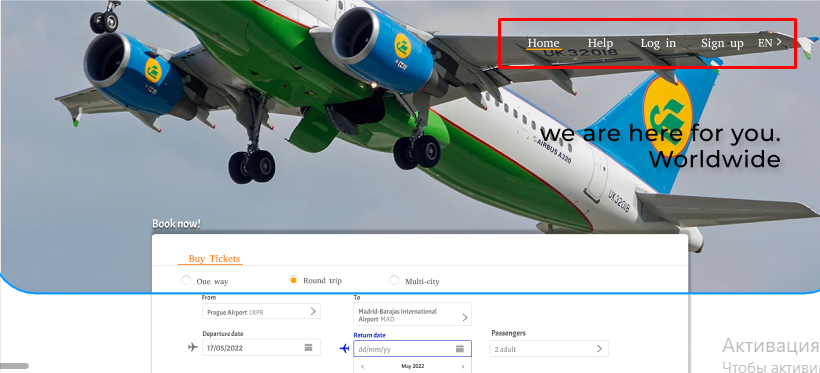


**Результаты остальных опытов внес в таблицу:**

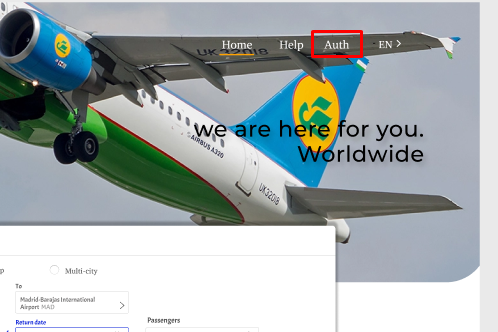


Как видно по графику при увеличение элементов время достижения тоже увеличиваятся. Но надо обратить внимание что при уменьшении элементов время уменьшается значительно с большой скоростью. Отсюда видно если у нас 1001 элементов, то нет смысла убрать одного, или даже 100 элементов. Но если у нас есть 5 элементов(как в моём случае), то можно сократить(если вожно) или выделить и т.п.





Давайте уберем **Login** и **Sign up**, а вместо этого используем **Auth** - думаю это слово уже набрало популярность как синоним Login и Sign up.



|  |  |
| --- | --- |
| Количество элементов для выбора | Время выбора(мс) |
| 4 | 404,8988547 |
| 5 | 445,6691876 |

При этом мы экономим 40мс жизни пользователя при **каждом** нажатии:)

**Практическая работа № 5.**

**Человеческие факторы. Закон Фитса.Факторы, влияющие на длительность физических действий пользователя.**

## **Закон Фитса**

Представим, что вы перемещаете курсор к кнопке, изображенной на экране. Кнопка является *целью* данного перемещения. Длина прямой линии, соединяющей начальную позицию курсора и ближайшую точку целевого объекта, определяется в законе Фитса как *дистанция*. На основе данных о размерах объекта и дистанции **закон Фитса** позволяет найти среднее время, за которое пользователь сможет переместить курсор к кнопке.

В одномерном случае, при котором размер объекта вдоль линии перемещения курсора обозначается как *S*, а дистанция от начальной позиции курсора до объекта — как *D* (рис. 6), закон Фитса формулируется следующим образом:

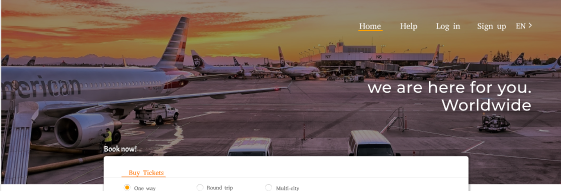
*Время (мс) = a + b log2(D/S+1)*

(Константы *a* и *b* устанавливаются опытным путем по параметрам производительности человека.)

Здесь a = 45, b = 155

**Начинаем опыт**

**Опыт 1**



Цель нажать на кнопку смены языка(на правом верхнем углу)

При этом S = 210мм, а D = 26мм.

Исходя из формулы:

*Время (мс) = a + b log2(D/S+1)*

T = 45 + 155\* *log2* (210/26+1) = 538мс = 0,538 с.

796px = 210мм



**Опыт 2.** Давайте теперь уменьшим расстояние до цели, до 5 мм

При этом S = 5мм, а D = 26мм.

Исходя из формулы:

*Время (мс) = a + b log2(D/S+1)*

T = 45 + 155\* *log2* (5/26+1) = 84,3мс = 0,084 с.



График изменения времени для достяжения нашей цели(если размер цели не меняется)

**Вывод: Чем больше размер и чем ближе расстояние, тем быстрее реакция пользователя, тем меньше время на достижения цели.**