



ПРОФСОУХ'17

# UI как иностранный язык

Ключ к пониманию пользователей

Одновал Евгений

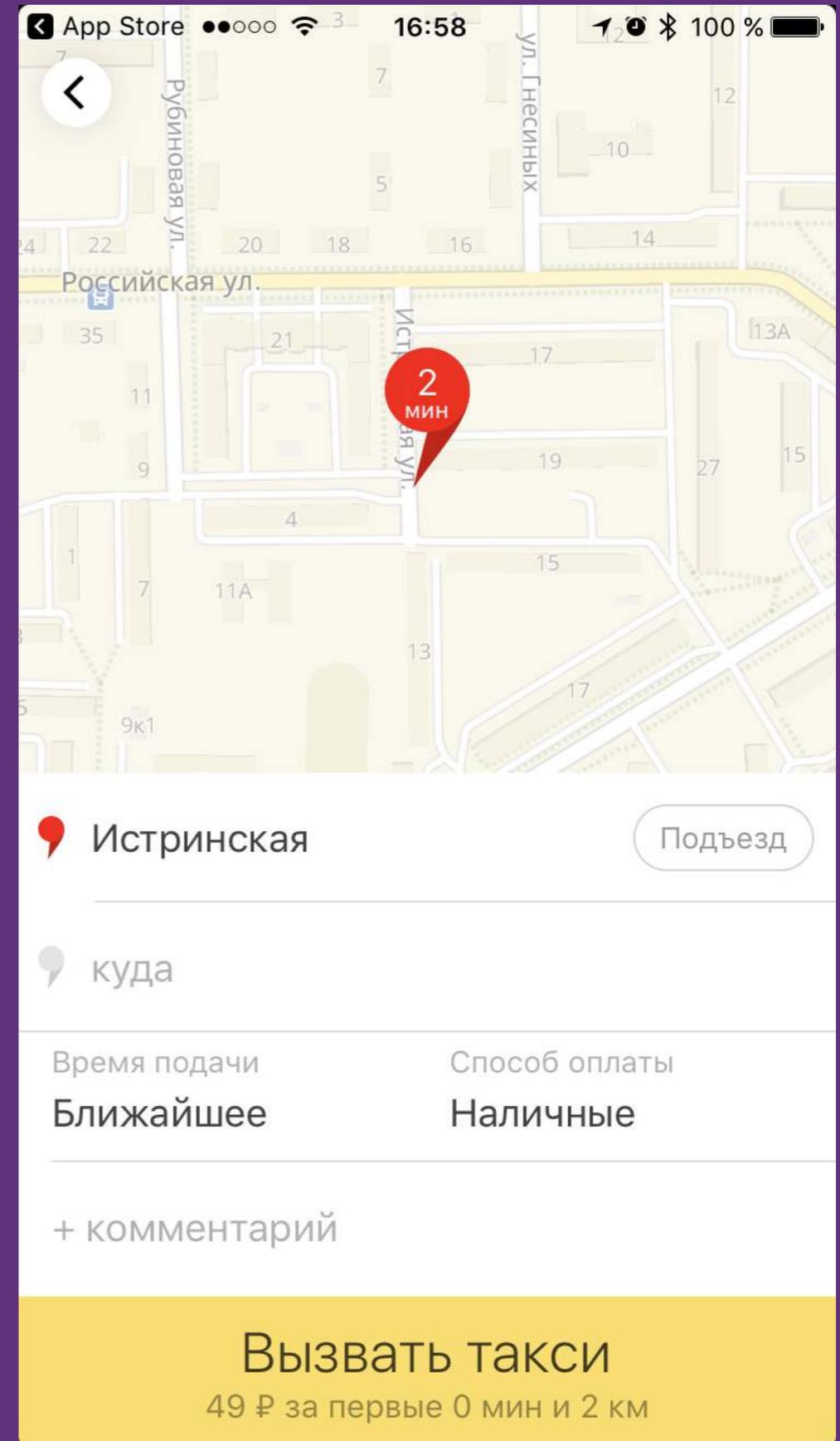
— мы проектировщики UI, что мы хотим создать?

— **ИНТУИТИВНО-ПОНЯТНЫЙ**  
пользовательский интерфейс для  
**(не)опытных** пользователей!



1

# Из чего складывается **понимание UI**



2

# Как **опыт** влияет на понимание



# UI как иностранный язык

Пользователь общается с сервисом средствами UI:

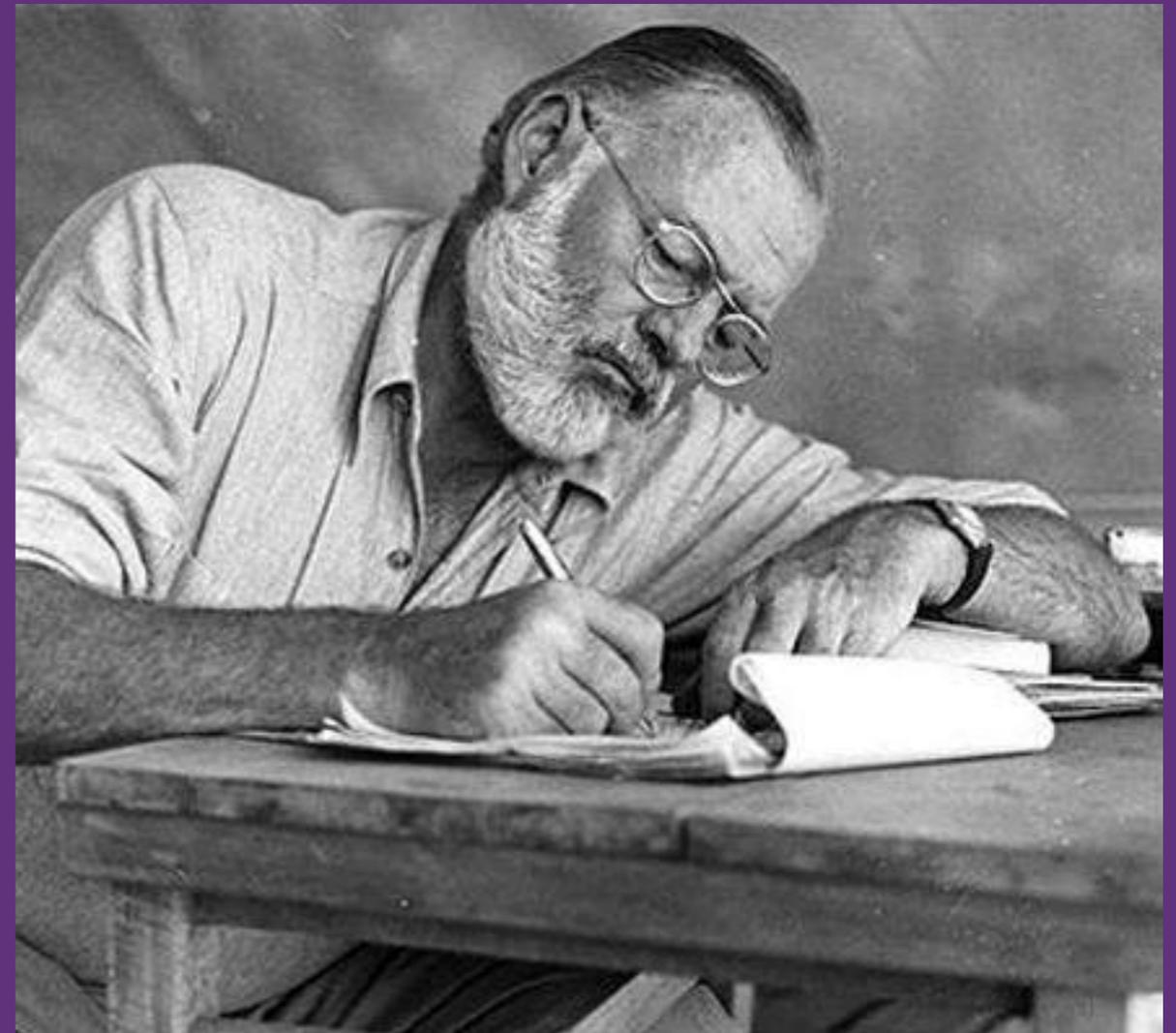
- **«Читает»** UI, чтобы понять, что делать
- **«Говорит»** сервису на языке UI о своих задачах и чувствах

# UI как иностранный язык

Проектировщик помогает пользователю решать задачи средствами UI:

- **«Пишет»** на языке UI сценарий
- **«Слушает»** ответы пользователя

Как научиться **писать**  
на языке UI с нуля



Как **не запутать**  
пользователей своей  
формой изложения



Как **развить UI** до  
уровня **native speaker**



# Понимание UI и ОПЫТ

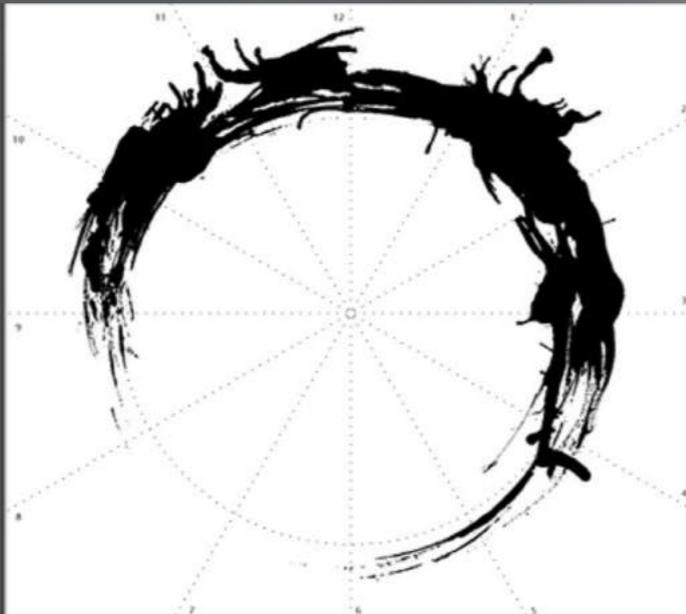
Пользовательский интерфейс —  
средство взаимодействия пользователя  
и объекта,

при помощи которого

пользователь **сообщает** объекту  
информацию о своих потребностях и  
**получает в ответ** информацию, что  
делать дальше.



Как человек научился  
пользоваться чайником?



1. Прочитал инструкцию

2. Возникло **понимание**:

«чтобы включить чайник, нужно  
поднять рычажок»

3. Нашел рычажок на чайнике

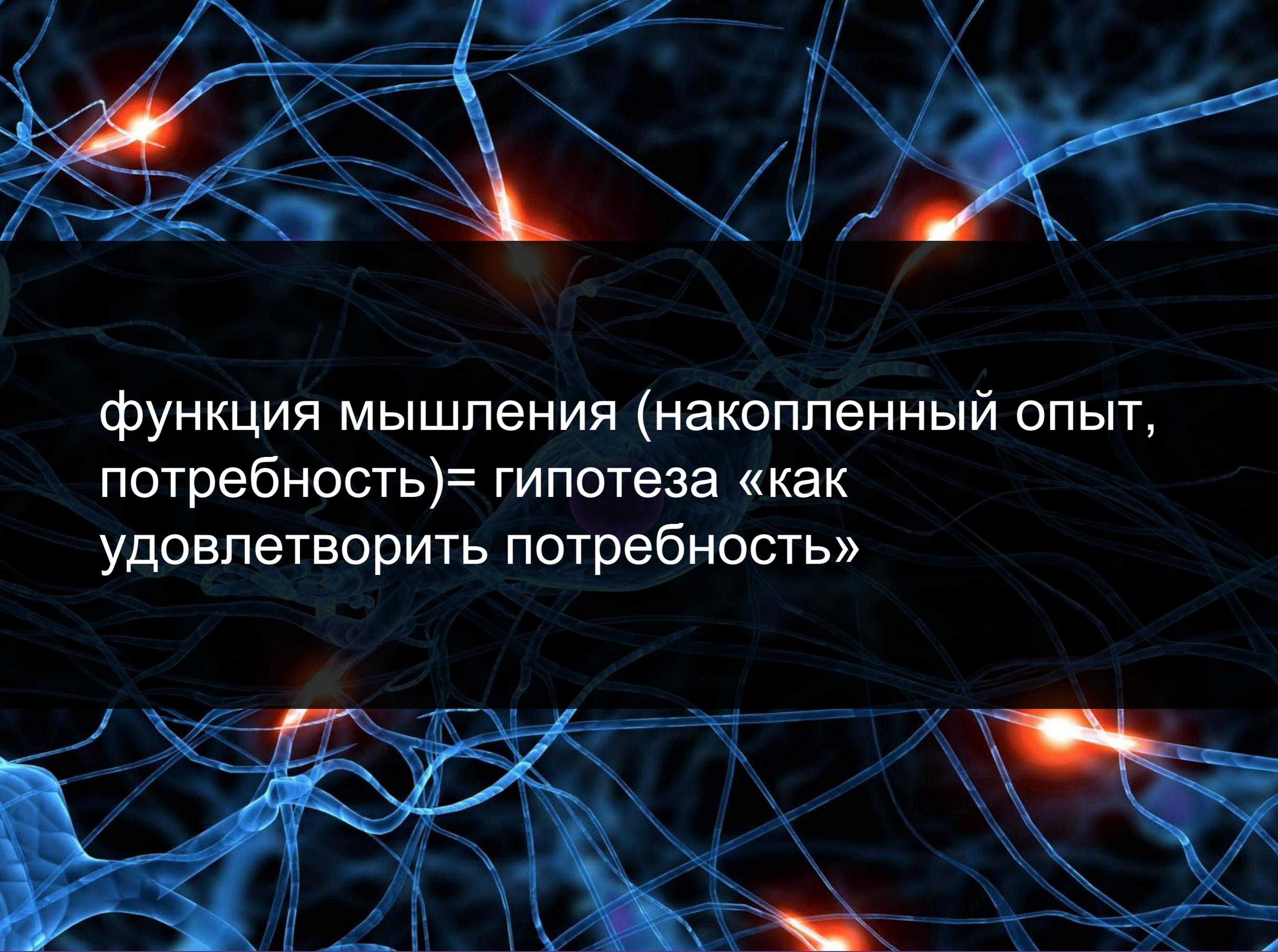
4. Поднял рычажок

5. Чайник включился

понимание > проверка > опыт

гипотеза > тестирование >  
фактический результат

Результат понимания UI —  
**предположение**, что нужно  
сделать, чтобы удовлетворить  
потребность

The background features a complex network of glowing blue lines, resembling neural connections or data pathways. Several nodes or junctions are highlighted with bright orange-red light, creating a sense of energy and activity. The overall aesthetic is futuristic and technological.

функция мышления (накопленный опыт,  
потребность) = гипотеза «как  
удовлетворить потребность»

50 оттенков  
(не)опытности

# Неопытный пользователь ЖИЗНИ



# Появление понимания

Пробует всё подряд

Фиксирует свои ощущения



Накапливает опыт

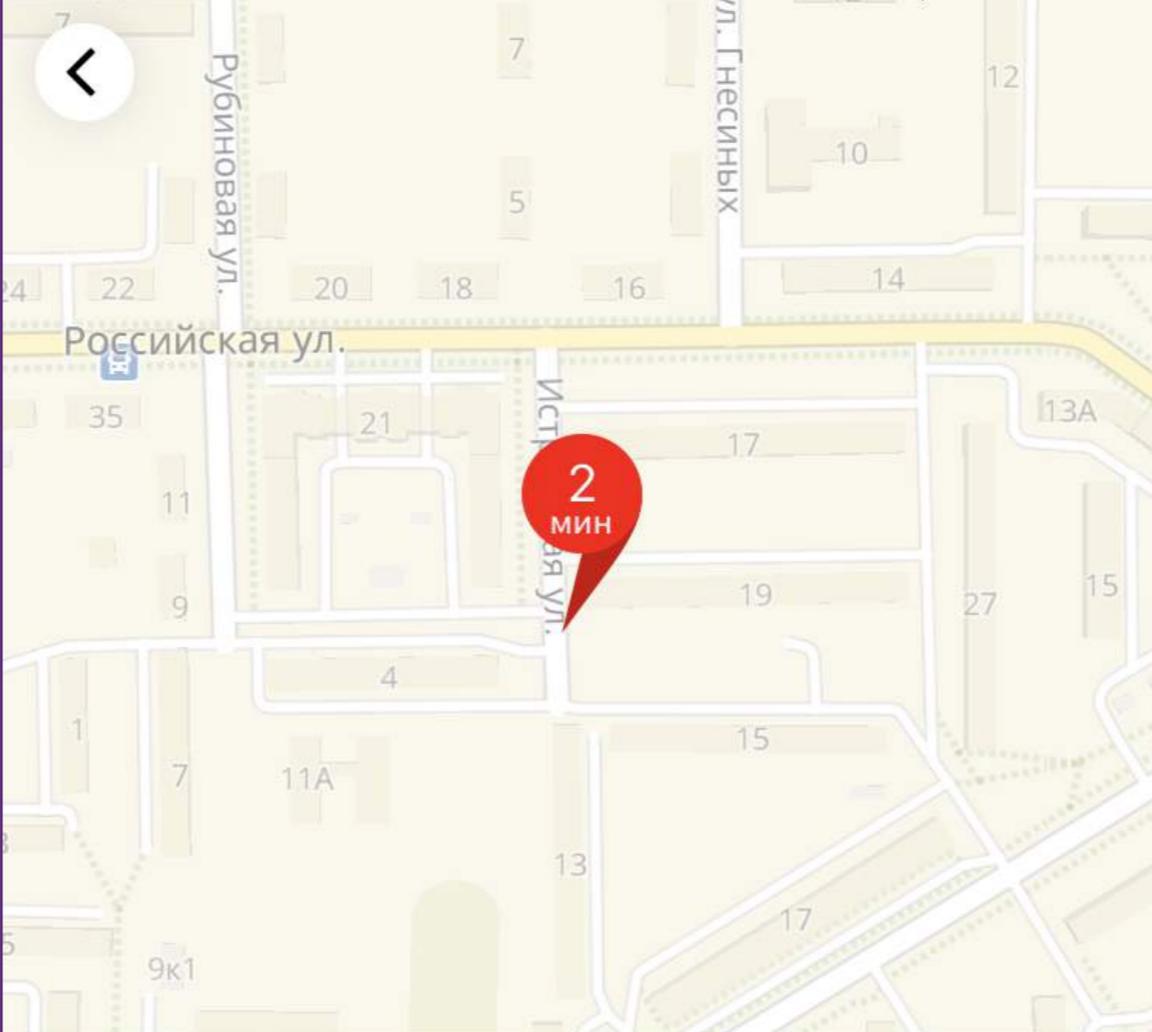
Развивает мышление





- Накопленный опыт  $> 0$
- Первый раз видят смартфон

Задача: заказать такси через приложение на смартфоне



Истринская

Подъезд

куда

Время подачи  
Ближайшее

Способ оплаты  
Наличные

+ комментарий

Вызвать такси  
49 Р за первые 0 мин и 2 км

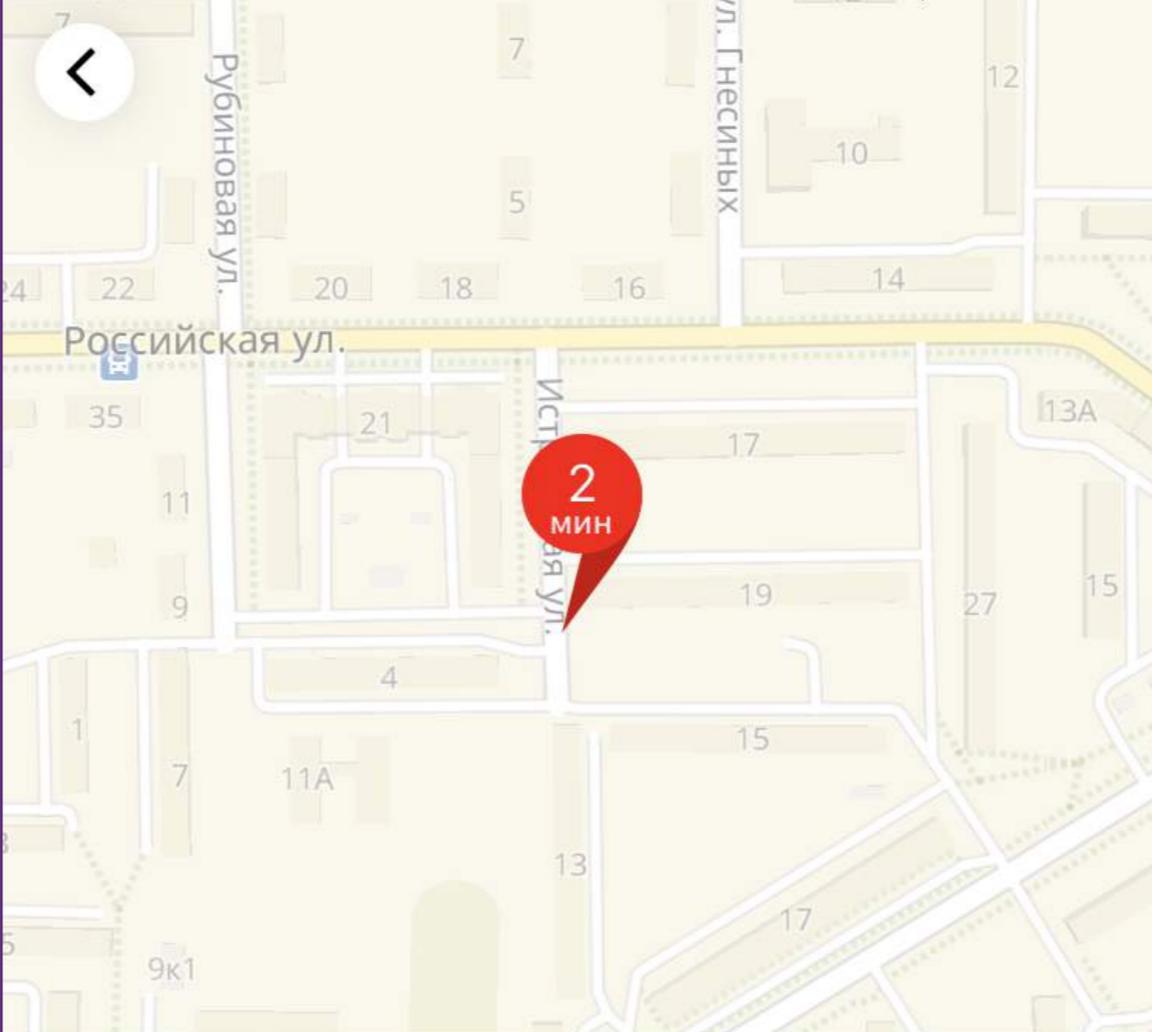


Разный аналоговый опыт

# Аналоговый опыт

формируется из

- культуры страны/города/района/дома
- воспитания и образования
- врожденных талантов и личных качеств
- образа жизни



Истринская

Подъезд

куда

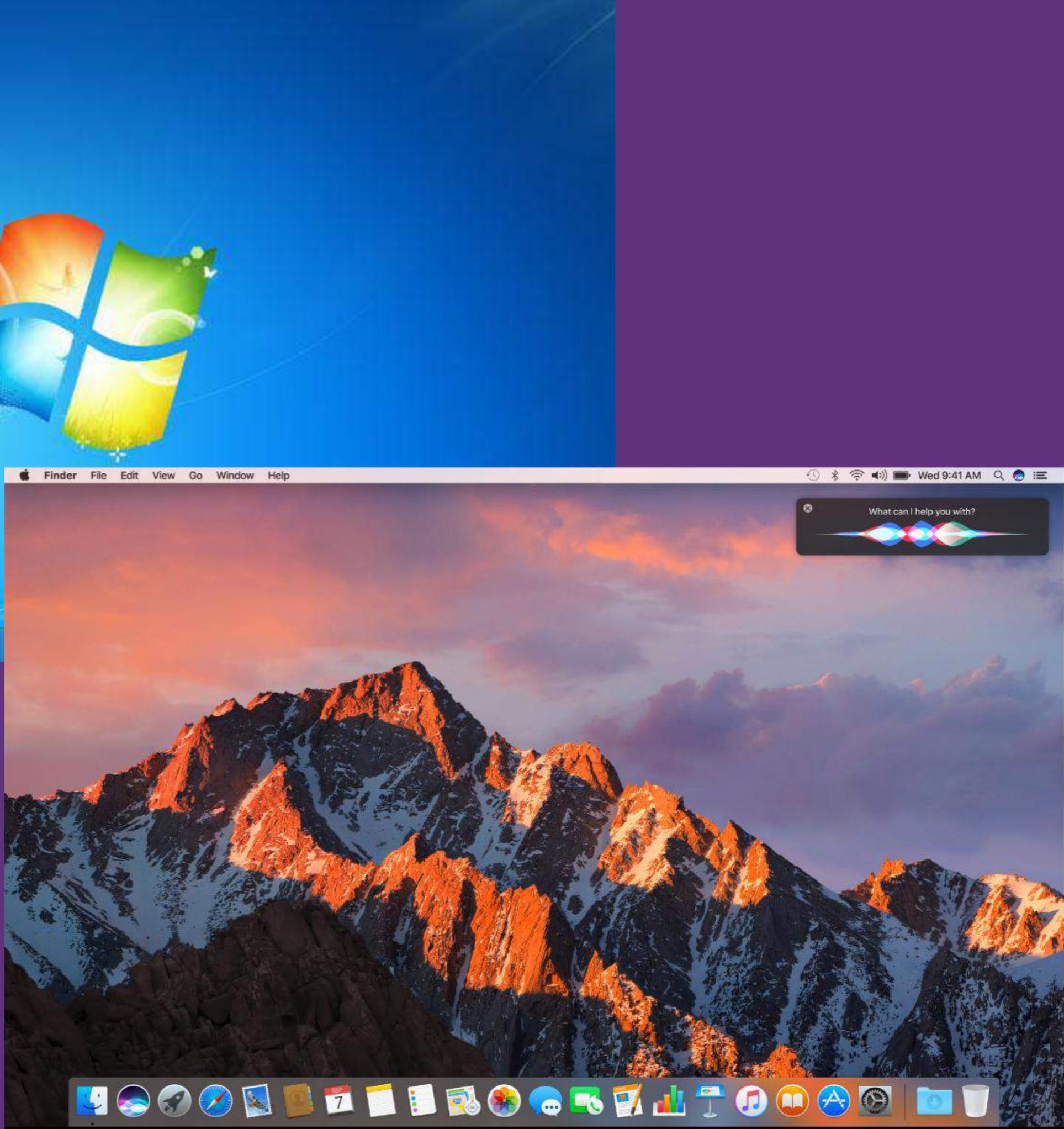
Время подачи  
Ближайшее

Способ оплаты  
Наличные

+ комментарий

Вызвать такси  
49 Р за первые 0 мин и 2 км

Цифровой опыт



# Цифровой опыт

формируется из

- опыта использования ОС (Win/Mac OS, iOS/Android/Win)
- опыта использования приложений
- прохождения обучения

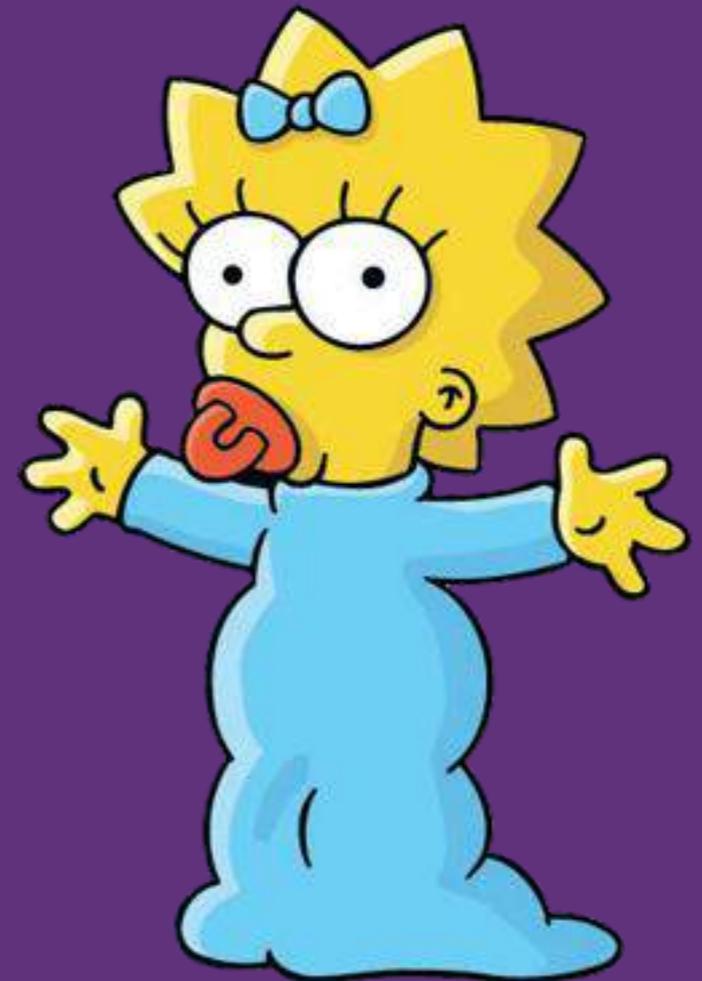
# Понимание UI

Функция мышления (UI, аналоговый опыт, цифровой опыт) = гипотеза «что сделать, чтобы удовлетворить свою потребность».

Разговариваем на UI

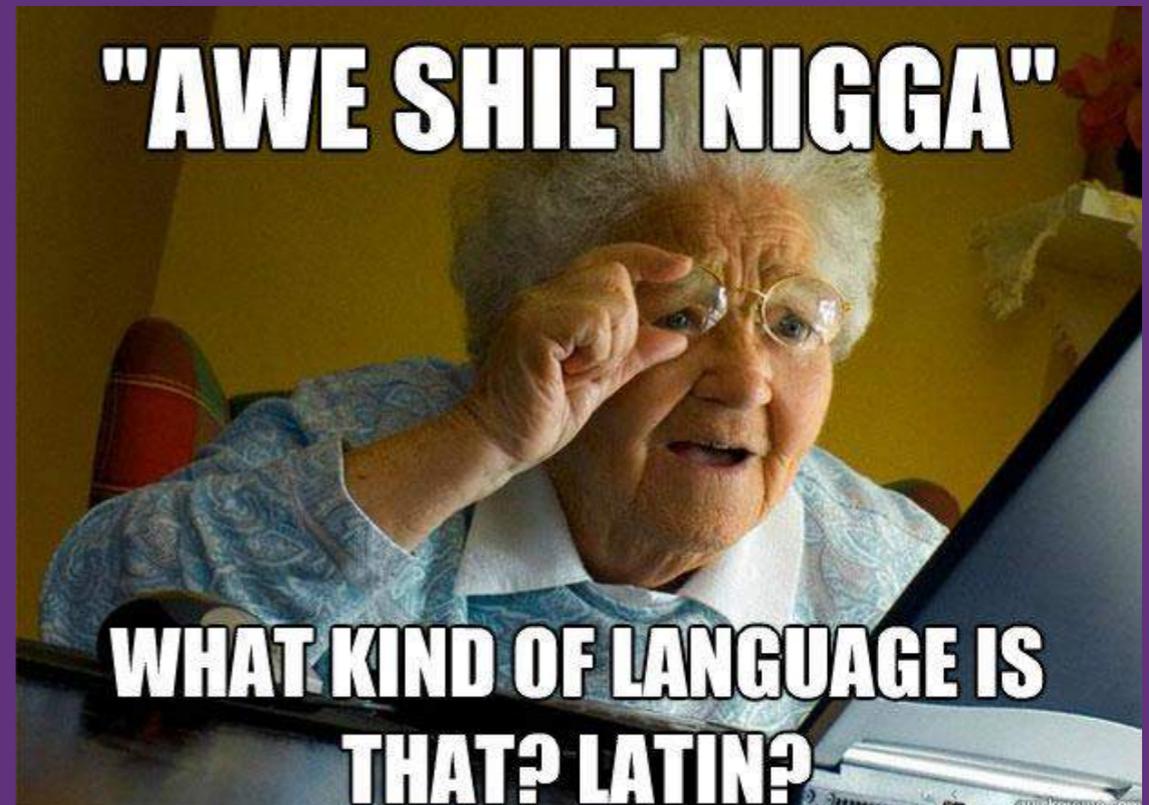
# Как выучить новый язык

- наблюдать, как общаются другие
- подражать
- методом проб и ошибок формировать навык коммуникации



**НАСЯЛЬНИКА**

**ПРИВЕТ**



С чего начать?

# UI разговорники

# iOS Human Interface Guidelines

The world's most advanced mobile OS offers everything you need to design beautiful, engaging apps that radiate power and simplicity.

Overview ▾

Design Principles

What's New in iOS 10

Interface Essentials

Interaction

Features

Visual Design

Graphics

UI Bars

UI Views

UI Controls

Extensions

Technologies

Resources



**Material design**

[Introduction](#)

[Environment](#)

[Material properties](#)

[Elevation & shadows](#)

[What's new](#)

**Motion**

**Style**

**Layout**

**Components**

**Patterns**

**Growth & communications**

**Usability**

**Platforms**

**Resources**

# Material design

## Introduction



Copyrighted Material

Steve Krug

UPDATED  
with three  
new chapters!

**DON'T  
MAKE  
ME**

**THINK**



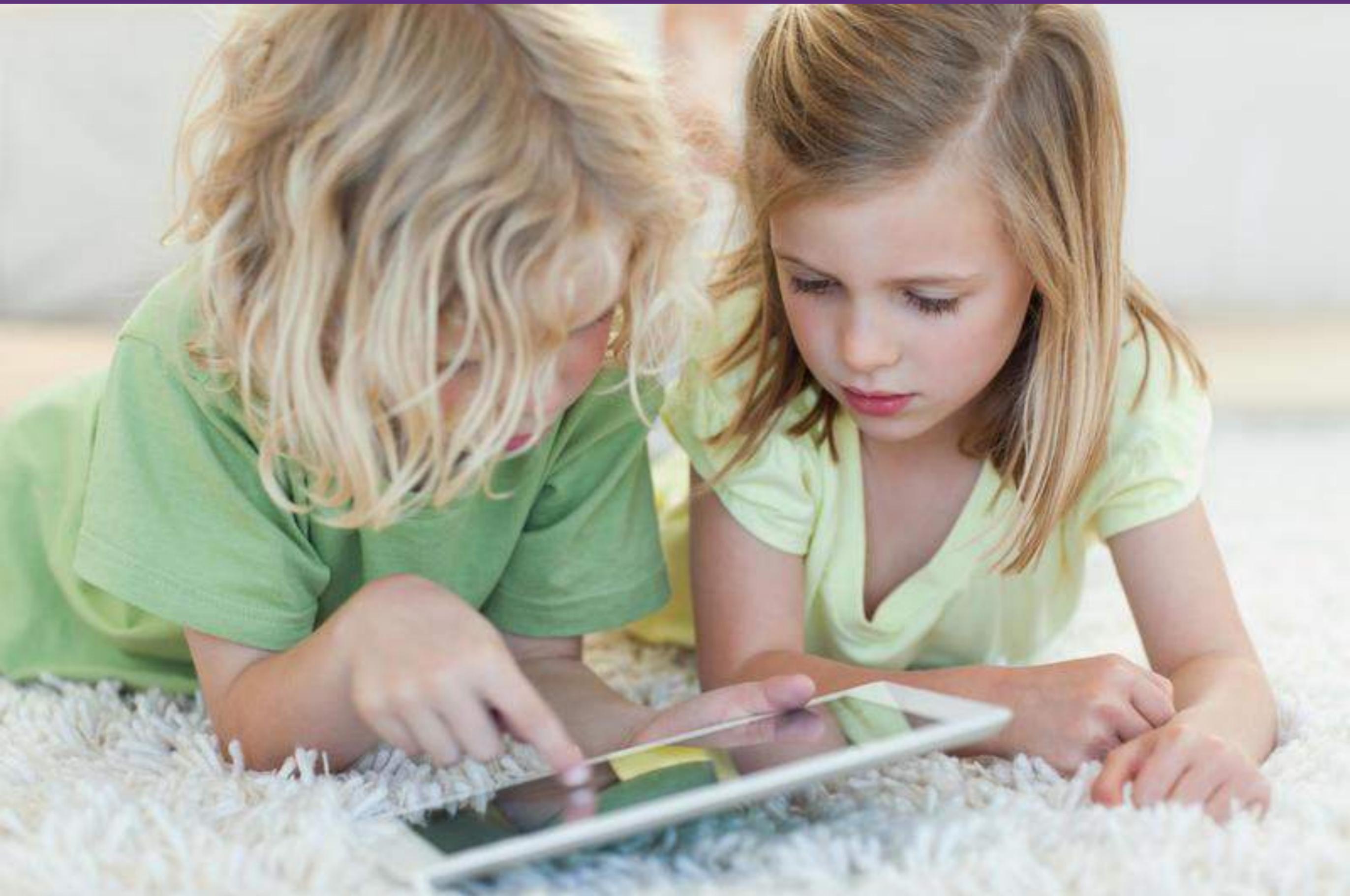
A Common Sense Approach to Web Usability

**SECOND EDITION**

Copyrighted Material

Существует ли панацея:  
легко-понятный UI элемент  
для всех?









Картинки



VS

СОХРАНИТЬ

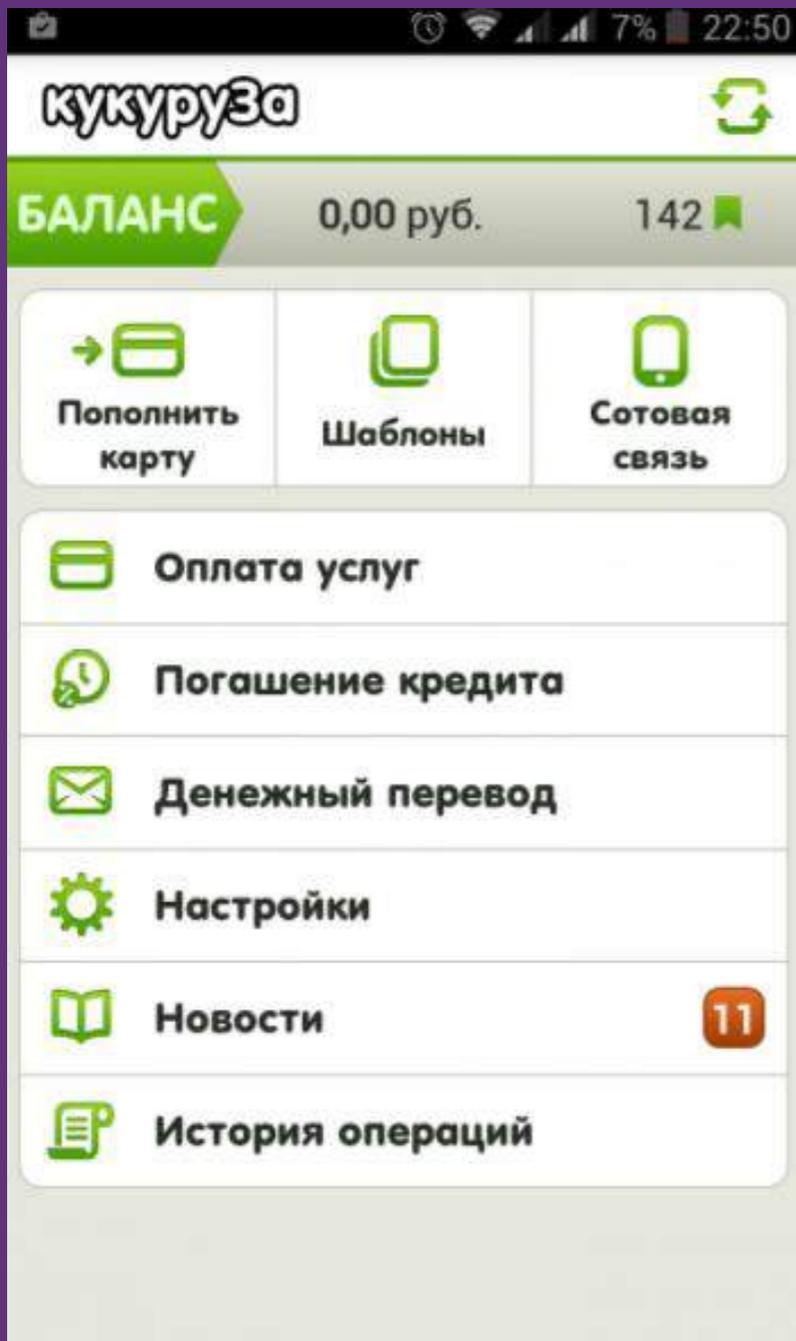
Текст

Текст — более точный связующий элемент смысла,

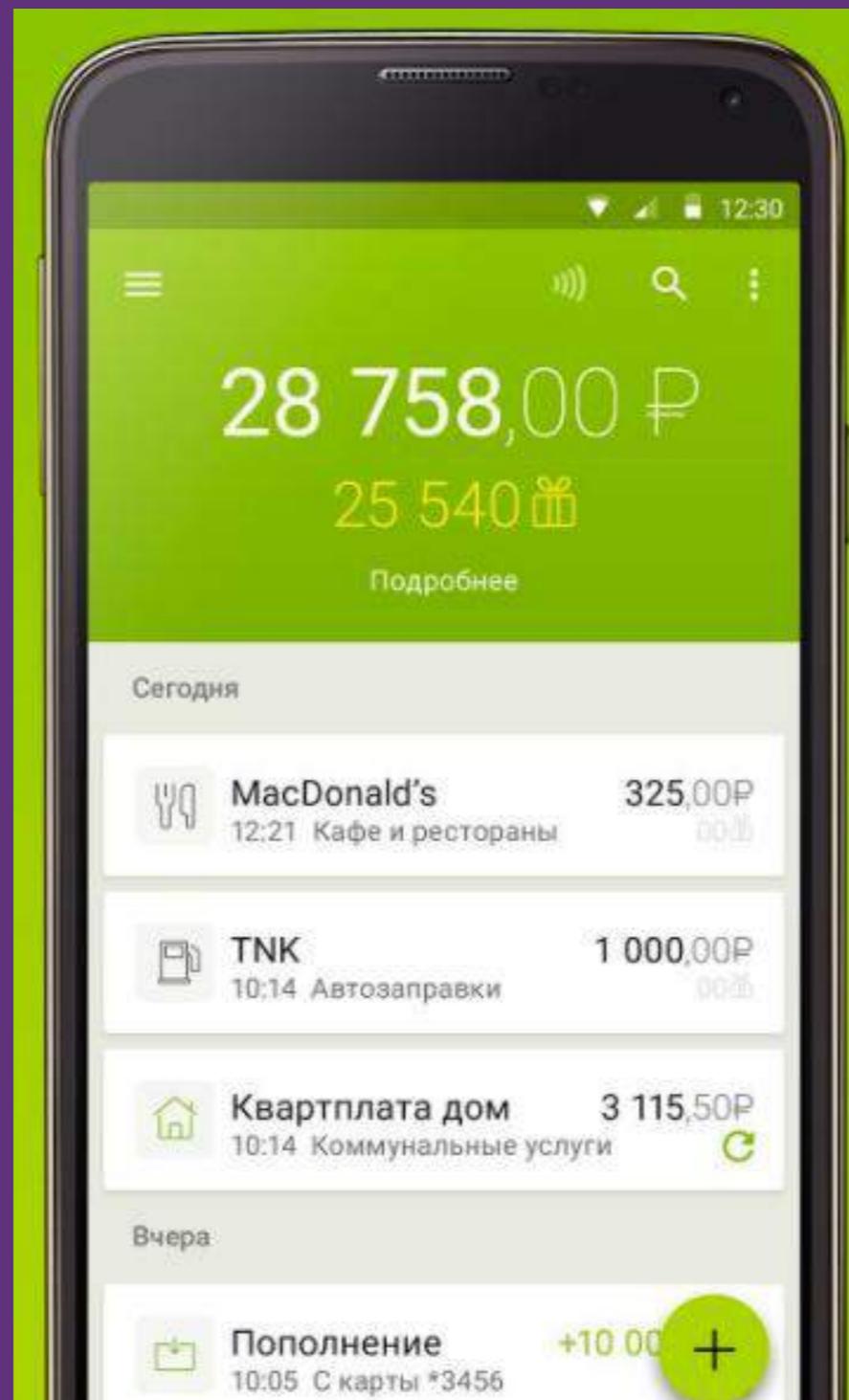
Картинка — более быстрый, но и более многозначный элемент смысла.

# Гамбургер? Не, не слышал.

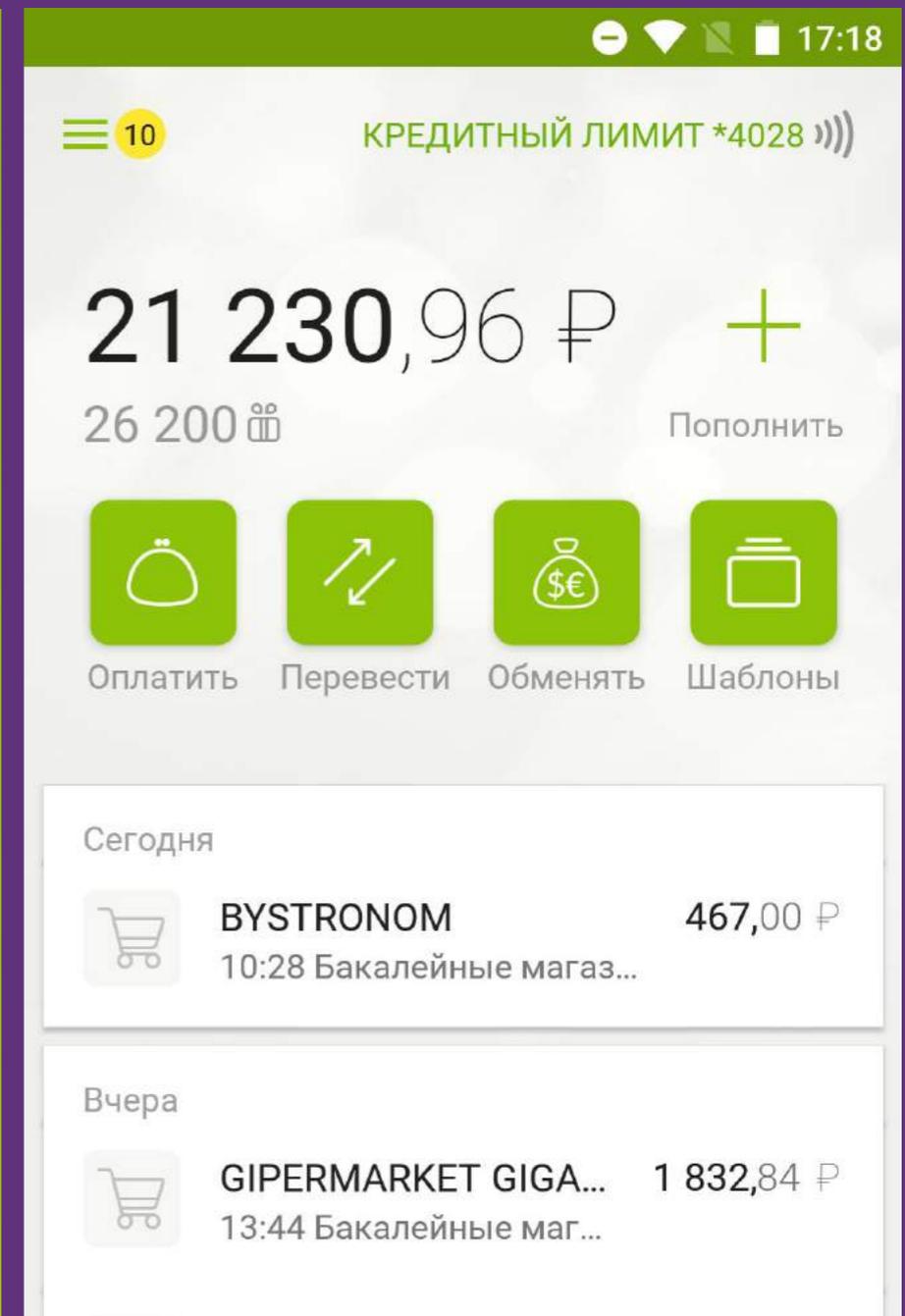
2013



2015



2016



Панацеи нет, но текст  
надежнее.

# Изучение языка

# Изучение языка

- тренировать фразы из разговорника
- читать литературу на иностранном языке
- разбирать непонятные выражения
- общаться с иностранцами
- преподавать

# Изучение языка

- тренировать фразы из разговорника
- читать литературу на иностранном языке
- разбирать непонятные выражения
- общаться с иностранцами
- преподавать



# Развитие языка UI

- быть опытным пользователем UI

# Развитие языка UI

- быть опытным пользователем UI
- учиться (книги, мастерклассы, и т.п.)

# Развитие языка UI

- быть опытным пользователем UI
- учиться (книги, мастерклассы, и т.п.)
- собирать обратную связь от ЦА

# Развитие языка UI

- быть опытным пользователем UI
- учиться (книги, мастерклассы, и т.п.)
- собирать обратную связь от ЦА
- корректировать гипотезы и снова их проверять

# Развитие языка UI

- быть опытным пользователем UI
- учиться (книги, мастерклассы, и т.п.)
- собирать обратную связь от ЦА
- корректировать гипотезы и снова их проверять
- преподавать

Native speaker UI?

# Программирование как иностранный язык

```
secretlevel = false;
else
    secretlevel = true;
gameaction = ga_completed;

void G_DoCompleted(void)
{
    if (gameaction == ga_completed)
        return;

    for (i=0; i<MAXPLAYERS; i++)
        if (!players[i].didsecret)
            players[i].didsecret = true;

    // take away cards and stuff
    for (i=0; i<MAXPLAYERS; i++)
        if (automapactive)
            AM_Stop();

    if (gamemode != commercial)
        switch(gamemode)
        {
            case B_PW:
                gameaction = ga_victory;
                return;
            case B:
                for (i=0; i<MAXPLAYERS; i++)
                    players[i].didsecret = true;
                break;
        }

    if (gamemap == B1)
        && (gamemode != commercial)

    if (gamemap == B9)
        && (gamemode != commercial)

    // draw the pattern into the back screen
    R_FillBackScreen();

    char name1[100];
    char name2[VERSIONSIZE];
    char* description;
    int length;

    if (M_CheckParm("-cham7"))
        sprintf(name1, "%s\\data\\SAVEGAMENAME%02d.dsg", savegameid);
    else
        sprintf(name1, "%s\\data\\SAVEGAMENAME%02d.dsg", savegameid);
    description = savedescription;

    save_p = savebuffer + savestringsize;

    memcpy(save_p, description, SAVESTRINGSIZE);
    save_p += SAVESTRINGSIZE;
    memcpy(name2, savedescription, VERSIONSIZE);
    sprintf(name2, "version %02", VERSION);
    memcpy(save_p, name2, VERSIONSIZE);
    save_p += VERSIONSIZE;

    *save_p++ = gameskill;
    *save_p++ = gameepisode;
    *save_p++ = gamemap;
    for (i=0; i<MAXPLAYERS; i++)
        *save_p++ = players[i].playerstate;
    *save_p++ = leveltime >> 16;
    *save_p++ = leveltime <> 16;
    *save_p++ = leveltime;

    P_ArchivedPlayers();
    P_ArchiveWorld();
    P_ArchiveThinkers();

    if (paused)
    {
        paused = false;
        S_ResumeSound();
    }

    if (skill > sk_nightmare)
        skill = sk_nightmare;

    // This was quite messy with SPECIAL and commented parts.
    // Supposedly hacks to make the latest edition work.
    // It might not work properly.
    if (episode < 1)
        episode = 1;

    if (gamemode == retail)
    {
        if (episode > 4)
            episode = 4;
    }
    else if (gamemode == shareware)
    {
        if (episode > 1)
            episode = 1; // only start episode 1 on shareware
    }
    else
    {
        if (episode > 3)
            episode = 3;
    }

    if (map > 9)
        && (gamemode != commercial)
        map = 9;

    ClearRespawners();
    respawmonsters = false;

    if (fastparm || (skill == sk_nightmare && gameskill == sk_nightmare))
    {
        for (i=S_SARG_RUN1; i<S_SARG_PWIN2; i++)
            states[i].tics >>= 1;
        mobinfo[MT_BRUISERSHOT].speed = 20*FRACUNIT;
        mobinfo[MT_HEADSHOT].speed = 20*FRACUNIT;
        mobinfo[MT_TROOPSHOT].speed = 20*FRACUNIT;
    }
    else if (skill != sk_nightmare && gameskill == sk_nightmare)
    {
        for (i=S_SARG_RUN1; i<S_SARG_PWIN2; i++)
            states[i].tics <<= 1;
        mobinfo[MT_BRUISERSHOT].speed = 15*FRACUNIT;
        mobinfo[MT_HEADSHOT].speed = 10*FRACUNIT;
        mobinfo[MT_TROOPSHOT].speed = 10*FRACUNIT;
    }

    // force players to be initialized upon first level load
    for (i=0; i<MAXPLAYERS; i++)
        players[i].playerstate = PST_REBORN;

    usergame = true; // will be set false if a demo
    paused = false;
    demoplayback = false;
    autoperspective = false;
    viewactive = true;
    gameepisode = episode;
    gamemap = map;
    gameskill = skill;
}

switch (gameaction)
{
    case ga_loadlevel:
        G_DeLoadLevel();
        break;
    case ga_newgame:
        G_DeNewGame();
        break;
    case ga_loadgame:
        G_DeLoadGame();
        break;
    case ga_savegame:
        G_DeSaveGame();
        break;
    case ga_playdemo:
        G_DePlayDemo();
        break;
    case ga_completed:
        G_DeCompleted();
        break;
    case ga_victory:
        F_StartFime();
        break;
    case ga_worlddone:
        G_DeWorldDone();
        break;
    case ga_screenshot:
        M_ScreenShot();
        gameaction = ga_nothing;
        break;
    case ga_nothing:
        break;
}

// get commands, check consistancy
// and build new consistency che
buf = (gameaction <> BACKL)

for (i=0; i<MAXPLAYERS; i++)
{
    if (players[i].cmd
        cmd = &players[i].cmd
        memcpy(cmd, &player

    if (demoplayback)
        G_ReadDemoTo();
    if (demorecording)
        G_WriteDemoTo();

    // check for turbo cheat
    if (cmd->forwardmove
        && !gametic & 31)

        static char turbo
        extern char *pla
        sprintf (turbo,
        players[i].consolep

    if (gametic > BA
        && consistanc

        {
            _Error ("cons
            cmd

        }

    if (players[i].no
        consistency)
        else
            consistency)
}

// check for special buttons
for (i=0; i<MAXPLAYERS; i++)
```



**Музыка как  
иностраннЫЙ язык**

Благодарю за внимание!

**Вопросы?**

[eugene.odnoval@gmail.com](mailto:eugene.odnoval@gmail.com)

[facebook.com/ejene](https://www.facebook.com/ejene)