Программный Регион

Golang application types

Application types

От типа приложения зависит его функционал и методы реализации.

- Демоны (daemon)
- Веб-приложения
- API

daemon

Демон Linux - это программа, у которой есть определенная уникальная цель. Обычно, это служебные программы, которые незаметно работают в фоновом режиме для того чтобы отслеживать состояние и обслуживать определенные подсистемы и гарантировать правильную работу всей операционной системы в целом.

daemon

Примеры в linux:

systemd - основная задача этого демона унифицировать конфигурацию и поведение других демонов в разных дистрибутивах Linux.

udisksd - обрабатывает такие операции как: монтирование, размонтирование, форматирование, подключение и отключение устройств хранения данных, таких как жесткие диски, USB флешки и т д.

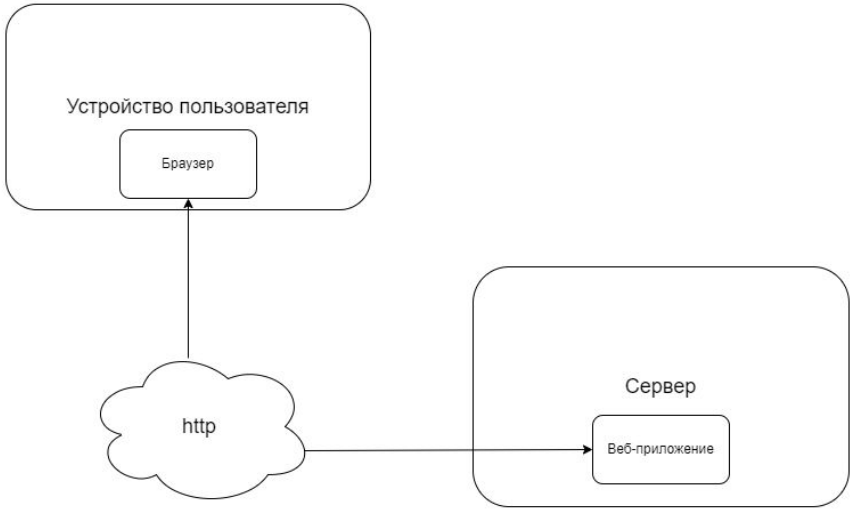
logind - небольшой демон, управляющий авторизацией пользователей.

Web application

Веб-приложение — клиент-серверное приложение, в котором клиент взаимодействует с веб-сервером при помощи браузера. Логика веб-приложения распределена между сервером и клиентом, хранение данных осуществляется, преимущественно, на сервере, обмен информацией происходит по сети.

Одним из преимуществ такого подхода является тот факт, что клиенты не зависят от конкретной операционной системы пользователя, поэтому веб-приложения являются межплатформенными службами.

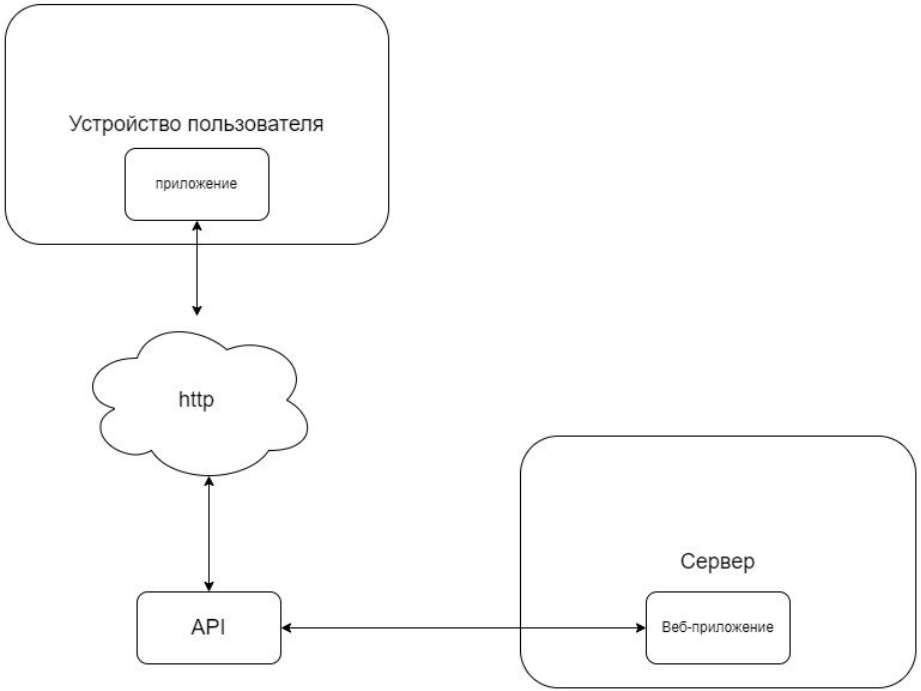
Web application



API

API (Application Program Interface) — это интерфейс прикладного программирования, который представляет собой набор готовых классов, процедур, функций, структур и констант, которые предоставляются сервисом для использования во внешних программных продуктах.

API


Устройство пользователя
приложение
http
API
Сервер
Веб-приложение

API

Быстрая регистрация в приложениях через аккаунты в социальных сетях.

Google. Эта система использует интерфейс программирования для предоставления разработчикам различных приложений доступа и возможности интеграции информации на разных сервисах. Например, можно найти и просмотреть видеоролик с платформы YouTube прямо в приложении.

Предоставление API в виде готового продукта. Разработчики предлагают доступ к своему приложению для получения оперативных данных по метеорологическим сводкам в любой точке земного шара и пр.

API

Для чего используют API:

- Упростить и ускорить выпуск новых продуктов, так как можно использовать уже готовые API для стандартных функций
- Сделать разработку более безопасной, выведя ряд функций в отдельное приложение, где они будут скрыты
- Упростить настройку связей между разными сервисами и программами

Simple hhtp sever

```go
package main

import (
    "fmt"
    "log"
    "net/http"
)

func main() {
    http.HandleFunc("/main", handler) // each request calls handler
    log.Fatal(http.ListenAndServe("localhost:8000", nil))
}

func handler(w http.ResponseWriter, r *http.Request) {
    fmt.Fprintf(w, "this is server URL.Path = %v\n", r.URL.Path)
}

```

Simple hhtp sever

- ServeMux - это мультиплексор HTTP-запросов. Он сопоставляет URL-адрес каждого входящего запроса со списком зарегистрированных шаблонов и вызывает обработчик для шаблона, который соответствует URL-адресу.
- HandleFunc - регистрирует функцию-обработчик для данного шаблона в DefaultServeMux.
- ResponseWriter - является интерфейсом, обеспечивающим доступ к методам, необходимым для возврата ответа клиенту.
- Request - это структура, содержащая проанализированное представление запроса от клиента.

Simple hhtp sever