

SAMSUNG



Kotlin

Базовый Курс

6.3. Передача данных по сети

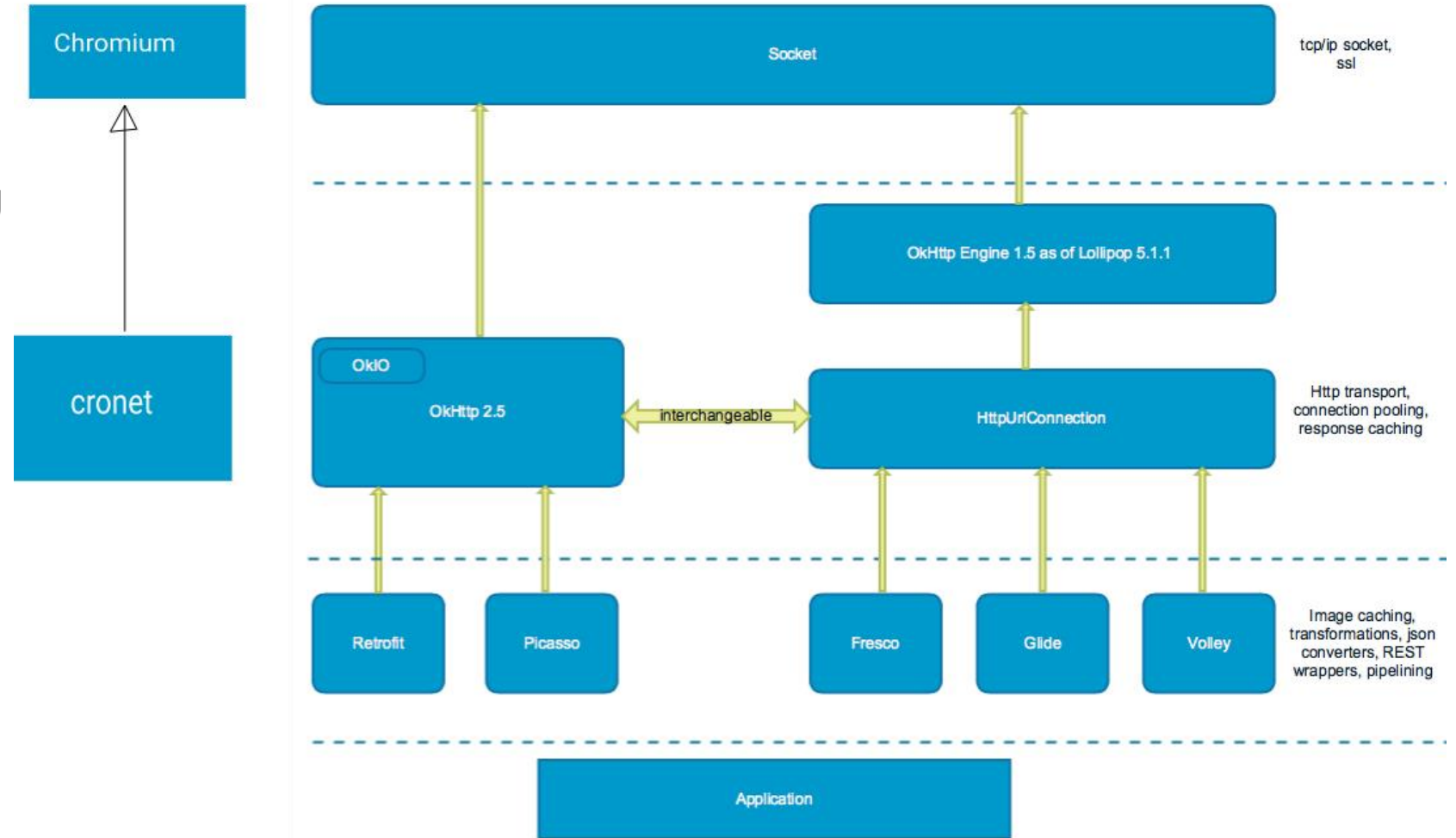


Сетевые фреймворки Android

SAMSUNG



- AndroidHttpClient
- OkHttp
- Fast Android Networking
- Voley
- Cronet
- Retrofit



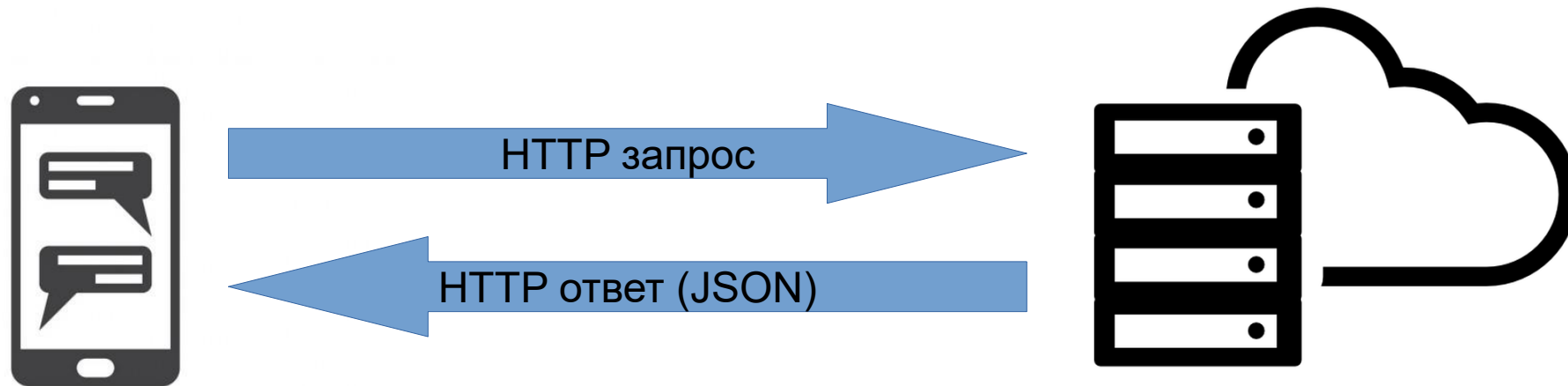


Сетевые фреймворки Android

SAMSUNG



Взаимодействие поверх HTTP





△ Retrofit, подключение

build.gradle (app)



```
implementation 'com.google.code.gson:gson:2.8.5'  
implementation 'com.squareup.retrofit2:retrofit:2.4.0'  
implementation 'com.squareup.retrofit2:converter-gson:2.4.0'
```

Манифест

```
<uses-permission android:name="android.permission.INTERNET" />  
<application  
    android:usesCleartextTraffic="true"  
    .....
```

SAMSUNG





Retetrofit, описание интерфейса

SAMSUNG



Описание WEB API (пример)

```
interface UserController {  
    @POST("/hello")  
    fun hello(@Body user: User): Call<Boolean>  
    @GET("/list")  
    fun list(): Call<List<User>>  
    // ...  
}
```



Аннотации методов

- GET,
- POST,
- PUT,
- PATCH,
- DELETE,
- HTTP,
- OPTIONS,
- HEAD.

Аннотации параметров

- @PATH,
- @QUERY,
- @BODY,
- ...





Использование Retrofit, код

SAMSUNG



Создание объекта Retrofit и сервиса:



```
val retrofit = Retrofit.Builder()
    .baseUrl(connectionURL)
    .addConverterFactory(GsonConverterFactory.create())
    .build()
val service: UserController = retrofit.create(UserController::class.java)
```



Подготовка HTTP запроса:

```
val call: Call<User> = service.get(firstname.text.toString(), lastname.text.toString())
```





Использование Retrofit, код

SAMSUNG



Async/Sync

```
val userResponse: Response<Boolean> =  
call.execute()  
val result: Boolean? = userResponse.body()  
Log.d("RetrofitClient", " " + result)
```

```
call.enqueue(object : Callback<User> {  
    override fun onResponse(call: Call<User>, response: Response<User>) {  
        if (response.isSuccessful) {  
            val user: User? = response.body()  
            result.text = "найден: " + user?.firstName + " " + user?.lastName  
        }  
    }  
    override fun onFailure(call: Call<User>, t: Throwable) {  
        Log.d("RetrofitClient", "Receive data from server problem ")  
    }  
})
```





Использование Retrofit, тест

SAMSUNG



Тестовый сервис:



- <https://httpbin.org/ip>
- <https://postman-echo.com/get?foo1=bar1&foo2=bar2>
- <https://putsreq.com/>
- `echo -e 'HTTP/1.1 200 OK\nContent-Length:10\n\n1111111111\n' | nc -NI 8000`
- `echo -e 'HTTP/1.1 200 OK\nContent-Length:34\n\n{"args":{"foo":"ccc","bar":"bbb"}}\n' | nc -NI 8000`

Json ↔ POJO

```
class MainData {  
    lateinit var args: Map<String, String>  
}
```

Или

```
class MainData {  
    lateinit var args: Args  
}  
class Args {  
    lateinit var foo: String  
    lateinit var bar: String  
}
```





Использование Retrofit, сервер

SAMSUNG



Код сервера:



```
@RestController
public class UserController {
    List<User> list=new LinkedList<User>();
```

```
    @RequestMapping(method = RequestMethod.POST,value = "/hello")
    public Boolean hello(@RequestBody User user) {
        System.out.println(user.firstName);
        list.add(user);
        return new Boolean(true);
    }
```

```
    @RequestMapping("/list")
    public List<User> list(){
        return list;
    }
```



}



```
public class User {
    public String firstName;
    public String lastName;
    public User(String firstName,String
lastName) {
        this.firstName=firstName;
        this.lastName=lastName;
    }
}
```





SAMSUNG



Спасибо

