AGH University of Science and Technology

5G network load generator

Android client

AGH

Section 1

Android traffic generator

Introduction

The traffic generator acting as a mobile user equipment is an android application capable of connecting to a central load testing server and performing requested tests. It listens for incoming tasks and executes them asynchronously. Resulting time measurments are aggreagted and uploaded to the server.

The core of the application is written in Kotlin.

Section 2

Android traffic generator functionality

Visualization

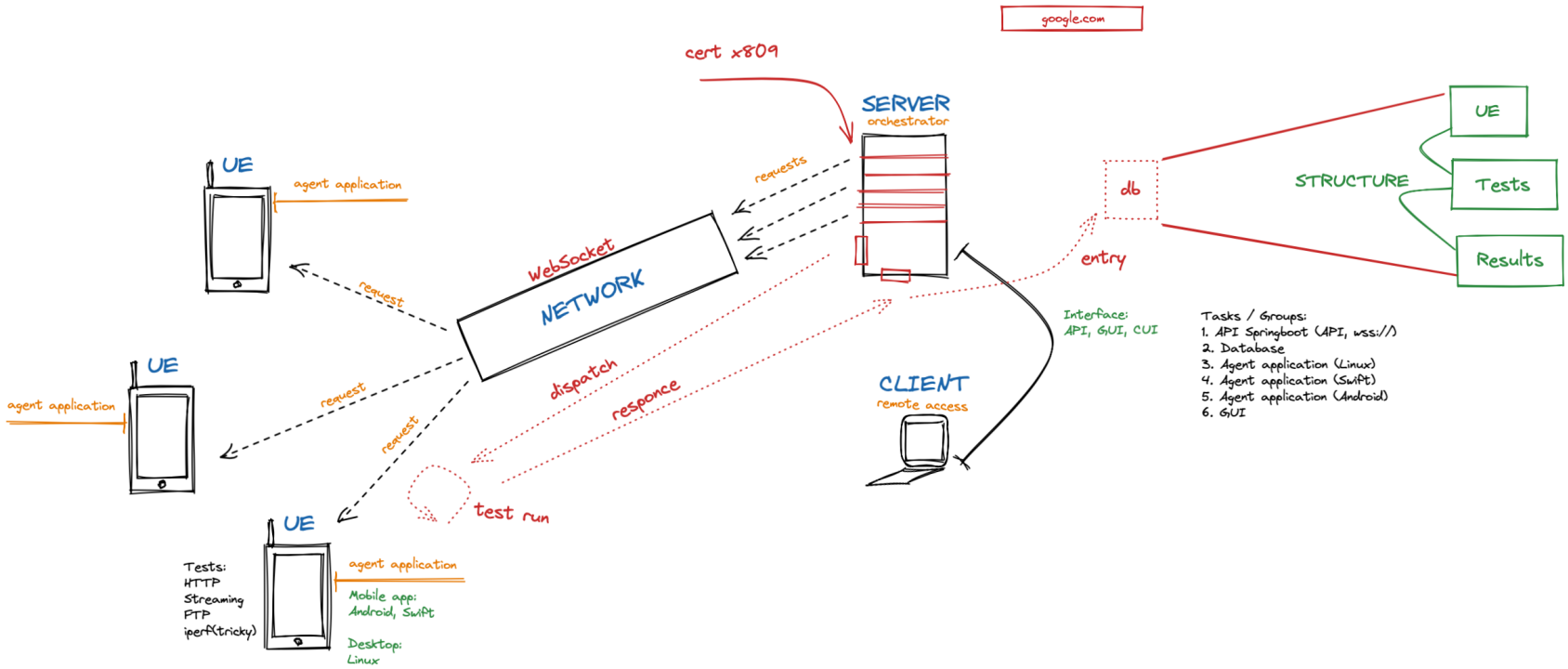


Figure 1: Process cycle diagram

Connection - authorization

Application sends login HTTP request and receives a JWT (`JSON Web Token`)

Related code can be found in `trafficgenerator.onActivityResult`.

authorization fields

- login
- password
- name or uuid (depending on whether device already exists)

Connection - heartbeat

After logging device into the system it is required to send HTTP POST request to `/device/active` url.

This is handled by a fixed rate timer identified as `"keepAlive"` that issues required request every 10 minutes and is cancelled upon logout.

related activities

- `trafficgenerator.onSuccessfulLogin`
- `trafficgenerator.logOut`

Retrieving tasks

After receiveing JWT client subsribes to a websocket that broadcasts notifications about new tasks being issued.

Subscribing to `/topic/device_${uuid}` url gives following message upon event:

websocket message

```
{
    "taskId": "long",
    "taskType": "string"
}
```

Retrieving tasks

Upon receiveing notification with `"taskId"` and `"taskType"` UE sends GET request to `/device/tasks` (`serverApi.getTasks`) in order to retrieve given tasks details. Look `trafficgenerator.newTaskReceived`.

Server responds with an array of all the tasks for a given device.

/device/tasks response

```
[
    {
        "id": "long",
        "taskType": "string",
        "status": "string",
        ...
```

Retrieving tasks

UE filters out all entries except for one with id received through websocket notification.

The task is added to the execution queue:

adding task to queue

```
if (tasks.isEmpty()) {
    appendStringToLog(
    "Failed to get task ${task.taskId} from all tasks"
    )
} else {
    asyncTaskExecutor.addTaskToExecutionQueue(
    tasks.first(), uuid, token, ::taskFinished
    )
}
```

Processing

Depending on "taskType" one of the scripts from trafficgenerator.scripts is executed.

available "scripts"

```
class httprequest(private val url: String){...}
class FtpClient(
    private val host      : String,
    private val port      : Int,
    private val userName  : String,
    private val password  : String,
    private val remoteCwd: String
)
private fun streamingTaskHandler(
    task: GetTasksResponseDTO)
```

Uploading results

Server provides endpoint `/device/tasks/${taskId}/upload` for uploading arbitrarily formatted files. It is used by application agent for uploading task results in a JSON file.

After task completion Android UE stores task duration and status in a file named `"task_${task.id}_result.json"` and sends it over to the server.

Uploading results

Results file is created by a JSON serializer - `GetTasksResponseDTO.Serializer()`

serialized data structure / class

```
data class GetTasksResponseDTO(
    val id: Long,
    val taskType: String,
    val status: String,
    val fileUrl: String?,
    val orderStart: String,
    val orderEnd: String?,
    val device: Device
) {
...  class Serializer { }
}
```

Disconnecting

Device is automatically logged out from the server after 10min of inactivity.

On the applications end `keepAliveTimer` is cancelled, connection handler closed and access token removed.

```
private fun logOut() {
    keepAliveTimer.cancel()
    serverApi.close()
    ...
    this.remove("token")
}
```

Login button is enabled so the connection might be reestablished:

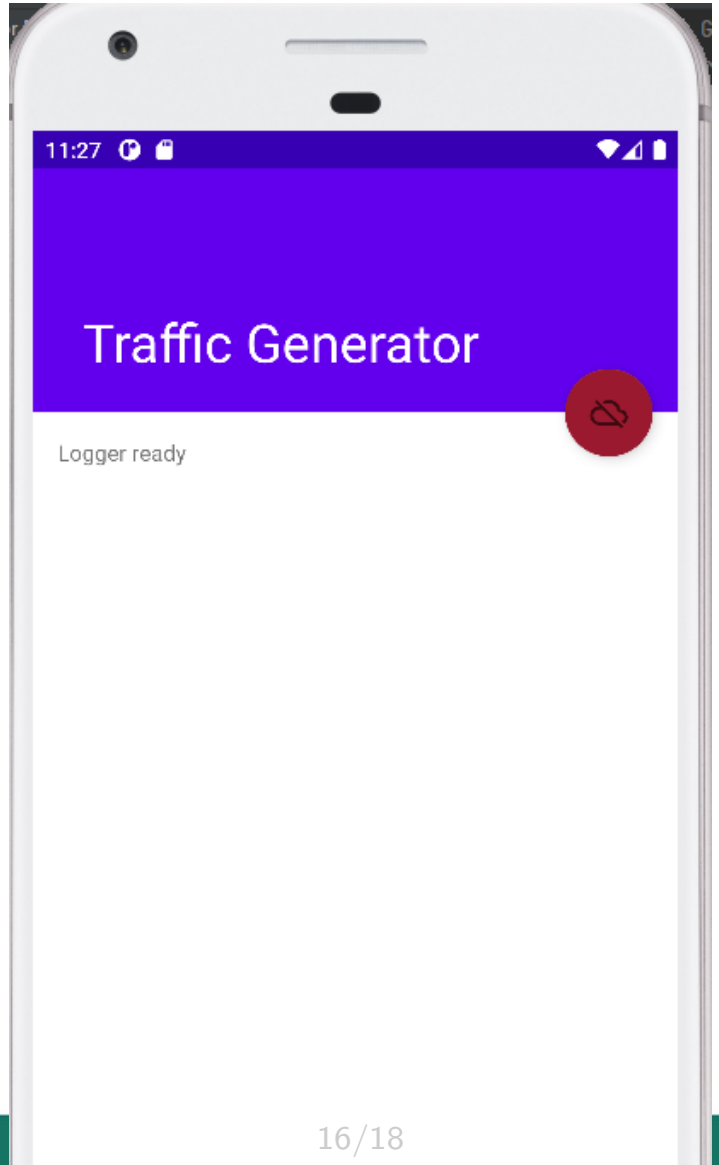
```
binding.fab.setOnClickListener{ openLoginActivity() }
```

AGH

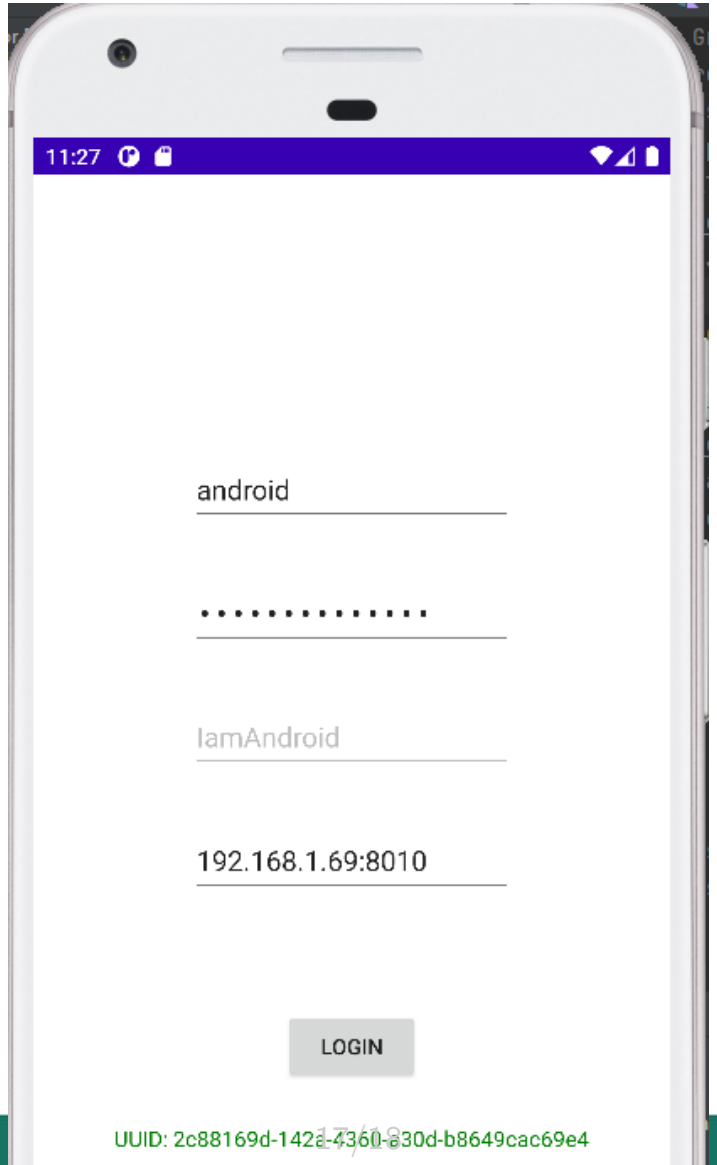
Section 3

Android App Simulation

First Screen



Login Screen



Screen with Logs