DroidLab keretrendszer

Az Android alapú elosztott kutatási hálózat tervei és prototípusának elemzése

Lajtha Balázs, Máté Miklós, Molnár Sándor, Mitcsenkov Attila, Vida Rolland

2013. január – március

# Bevezetés

Az okostelefonok és táblagépek térnyerése gyökeresen megváltoztatta az informatika világát. Az informatikus közösségben a hordozható okos eszközök csak egy újabb kelléket jelentenek a workstation és a laptop mellett ám a lakosság körében azokat leváltva terjednek, hiszen képesek kielégíteni a tartalomfogyasztási igényeket, olcsó, hordozható és kényelmes formában. A PC piac folyamatosan zsugorodik, egyes szegmensek teljesen eltűnnek.

A mobilizálódó világ új kihívások elé állítja a kutatókat és fejlesztőket. A vezetékes korszakhoz képest a szabad paraméterek száma megnőtt, az értékkészletük kiszélesedett így a hálózati topológiák, protokollok hagyományos vizsgálatában használt szimulációs megközelítések egyre korlátozottabban alkalmazhatóak. A szimulációk kiváltására életre hívott elosztott teszthálózatok, mint a PlanetLab vagy SenseLab sem tudják megfelelően imitálni a mobil felhasználók által létrehozott hálózati közeget. Ez a felismerés hívta életre a DroidLab Android alapú elosztott kuatató-környezetet.

A DroidLab rendszer nagy előnye, hogy nem laboratóriumi hardware-eket vesz igénybe. A rendszerhez csatlakozó felhasználók ajánlanak fel erőforrásokat, ezekből gazdálkodhatnak a kutatók. Ez a crowdsourcing megközelítés sok előnye mellett megkötésekkel is jár. Ebben a tanulmányban bemutatjuk a DroidLab keretrendszert. Az első fejezetekben a keretrendszerrel szemben támasztott igényeket és azokat a megkötéseket ismertetjük, melyeket a rendszer környezete teremt. Ezt követi az üzleti modell és a felhasználókat hozzájárulásra ösztönző eszközök bemutatása. A rendszer felépítését a 3. fejezetben tárgyaljuk. A tanulmányt az eddig elkészült prototípusból levont tanulságokkal és a további feladatok meghatározásával zárjuk.

# A Droidlab környezete

A DroidLab nem egy izolált, laboratóriumokba telepíthető hálózat, melyen kutatók időszeleteket bérelhetnek. Mélyen épül be a világba. Ezért a DroidLab tervezésekor a technikai részleteknél hangsúlyosabb a rendszer integrációja. Ehhez meg kell ismernünk azokat, akik a részt fognak venni a rendszerben. Bár a rendszer szó szoros értelemben vett felhasználói azok a kutatók, akik a rendszer segítségével méréseket végeznek, mi a felhasználók kifejezést azokra értjük, akik eszközeikkel vesznek részt a hálózatban. A rendszert használó kutatókra kutatóként fogunk hivatkozni. Annak ellenére, hogy a rendszer használatához a kutatóknak is programozniuk kell, fejlesztőnek azokat fogjuk nevezni, akik a rendszeren végeznek fejlesztéseket. A rendszer moduláris felépítéséből és az elvégzendő feladatok jellegéből adódóan a fejlesztők és a kutatók személyükben megegyezhetnek, de a különböző szerepekre azoknak megfelelően fogunk hivatkozni.

## 2. 1. Eszközök

A DroidLab egy központi vezérlőből és a hálózatban részt vevő felhasználók eszközeiből tevődik össze.

### Felhasználói eszközök

A DroidLab hálózatot kizárólag Android eszközökre valósítjuk meg. A választásunk három okból esett az Android platoformra. Egyrészt ma ez a platform a legelterjedtebb és Android eszközökből értékesítenek a legtöbbet, ezért a jövőben az Android platform részaránya még a mostaninál is magasabb lesz. Másfelől az Androidos eszközpaletta a legszínesebb. Karóráktól telefonokon és tableteken keresztül egészen televíziókig minden méretben megtalálható szórakoztatóelektronikai cikkekben, és beágyazott rendszerekben is terjed, így sokféle felhasználói viselkedésminta gyűjthető Android eszközökről. Az Android platform választása mellett szól az Android SDK széles szolgáltatás-spektruma is.

A rendszer által használt eszközök maradéktalanul olyan eszközök lesznek, melyeket tulajdonosaik elsősorban más feladatokra használnak. Elsősorban mobil eszközökre számítunk, a felhasználói mozgásból származik a DroidLab-on futtatott mérések igazi értéke. Ezzel azonban sok megkötés is jár. Az Android eszközök üzemideje még mindig érzékeny téma. Nem engedhető meg, hogy a rendszer ezt az üzemidőt jelentősen csökkentse.

### Központi vezérlő

A központi vezérlő nem vesz részt a mérésekben, csak azoknak a lebonyolítását végzi, ha valamelyik méréshez rögzített szerverek szükségesek, azokról a mérést végző kutatóknak kell gondoskodniuk, akár saját hardware, akár valamely más mérőhálózat pld. PlanetLab használatával. Ez a megközelítés csökkenti a vezérlő erőforrás-igényét, megkönnyítve a rendszer skálázhatóságát, és nagyobb szabadságot ad a kutatóknak méréseik megvalósításában.

## 2. 2. Felhasználók

Bárki részt vehet a rendszerben, aki Androidos eszközzel rendelkezik, ám értékes méréseket elsősorban a fiatalabb korosztály tud végezni, azon belül is a felsőoktatásban tanulók. Ebben a csoportban a legintenzívebb az interakció. A szabadabb napirend nagyobb mozgásteret biztosít, a rutin nem napi hanem heti ciklusú így nem annyira monoton. És az egyetemisták nem csak munkaidőben de szabadidejükben is mozgalmasabb életet élnek, rövidebb időbe sűrítik az egyéb felhasználói csoportok viselkedését így rövidebb idő alatt sokkal többféle minta figyelhető meg ebben a csoportban mint akár a középiskolások, akár a dolgozó vagy nyugdíjas felnőttek közt. Ezek a felhasználók alaposabb technikai ismeretekkel rendelkeznek, és nyitottabbak az új kezdeményezésekre. Így könnyebben motiválhatóak részvételre.

A projekt kezdeti szakaszában a laborban dolgozók és a labornak önálló feladatokat végző diákok adják majd a felhasználói bázist, ezt fogjuk fokozatosan kiterjeszteni az általunk oktatott diákokra majd a rendszer élesítése után a bevett csatornákon népszerűsítve terjesztjük az alkalmazást.

A rendszer a felhasználók által felajánlott erőforrásokra épít. A Seti@Home projekt világsikere bebizonyította, hogy ez egy használható modell: a felhasználók szabad erőforrásaikkal járulnak hozzá egy az emberiség számára fontos célhoz. Android környezetben azonban a felhasználók nem annyira bőkezűek.

Számolni kell sok olyan korlátozással, mely PCn nem jelent meg. A mobil eszközök szűkös tápellátása szűkíti az alkalmazási területet. A DroidLab nem egy elosztott számítási platform, hanem inkább szenzorhálózatként kell tekinteni rá. Hiába vannak a modern eszközökben PC-vel összemérhető teljesítményű processzorok, azoknak a kihasználása elfogadhatatlan mértékben csökkentené a hasznos üzemidőt. Kisebb teljesítményű készülékeknél pedig azért kell a processzor-használattal takarékoskodni, mert a terhelt rendszer a felhasználó szemszögéből lassú lesz, ami egy telefonnál nem elfogadható.

Szintén szűkös erőforrás a hálózati sávszélesség. Hazánkban az okostelefon-tulajdonosok fele nem rendelkezik mobil-adatszolgáltatással, a forgalmazott tabletek többsége pedig nem is képes cella alapú hálózatokhoz csatlakozni. Ezért a keretrendszert úgy terveztük meg, hogy állandó hálózati kapcsolat nélkül is képes legyen működni.

Az elmúlt években előtérbe került a személyes adatok védelme is. A felhasználók sokkal érzékenyebbek az okostelefonjukon tárolt adatok kompromittálódására mint a PC-jükön tárolt adatokra - még ha ezek jelentős átfedésben is vannak. Az okostelefon egy nagyon személyes tárgy, így a platform megvalósításakor fontos szempont az anonimitás megvalósítása.

Mindhárom területen elsődleges a megfelelő kommunikáció. A felhasználók nem rendelkeznek megfelelő szakértelemmel ahhoz, hogy eldöntsék hogy biztonságban vannak-e, ezért az alkalmazás felületének a biztonságot és a felhasználói kontrollt kell hirdetnie.

### Felhasználói élmény

A DroidLab felhasználói interakció nélkül is képes működni. Lehetséges út lett volna rejtőzködő megközelítést választani, eldugni az alkalmazást a felhasználó elől, minimálisra korlátozva a felhasználó és a rendszer közti interakciót. Mi nem ezt a megközelítést választottuk.

Célunk hogy az alkalmazás minden részletével részvételre buzdítsa a felhasználót. Ezért az alkalmazás azon túl hogy lehetővé teszi a mérések futtatását be fogja vonni a felhasználót a munkába. Ennek két oldala van. Az egyik hogy a telefon erőforrásait a felhasználó aktív részvételével foglalja a keretrendszer.

Az Android platform nem biztosít lehetőséget az alkalmazások erőforrás-használatának korlátozására. Az alkalmazások telepítésekor kérdezi meg a felhasználót arról ,hogy milyen jogosultságokat engedélyez az alkalmazásnak. Ezek a jogok finom granularitással szabályozzák a különböző adatokhoz és hardware-ekhez való hozzáférést. Ám ezek csak a hozzáférés minőségét szabályozzák, nem a mennyiségét, és azt sem mindig elégséges szinten. Az *Internet* jogosultság például korlátlan hálózati adatforgalmat biztosít, tekintet nélkül a használt hálózati technológiára ami a gyakorlatban igen kockázatos vállalkozás hiszen a cella alapú adatforgalom a felhasználónak pénzébe is kerülhet. Sokan bírálják az Android jogosultságkezelési rendszerét amiatt is, hogy a telepítéskor egyben kell elfogadni vagy elutasítani a különböző jogosultságokra vonatkozó kérelmeket, azokat nem lehet futási időben elvenni vagy utólag jóváhagyni. Ez a megközelítés egy olyan alkalmazásnál mint a DroidLab, mely a felhasználó komfortzónáját figyelembe véve használná a telefont kifejezetten rossz üzenetet közvetít. Ennek a problémának a megkerülésére a későbbiekben ismertetjük a kidolgozott megoldást.

Azon túl, hogy a felhasználónak lehetőséget adunk a felajánlott erőforrások testre szabására mind minőségi, mind mennyiségi szempontból, úgy próbáljuk bevonni a munkába, hogy tájékoztatjuk az éppen futó mérések részleteiről és kivonatosan jelentjük felé is a mérések során elért eredményeket. Ennek a megvalósításában a kutatók játszanak fontos szerepet.

#### Felhasználói motiváció

Azon túl hogy rálátást adunk a felhasználónak a futó mérésekre és irányító szerepet adunk neki a felhasznált erőforrások tekintetében a kész rendszerbe egyéb ösztönző mechanizmusokat is implementálni fogunk. A célzott korosztály fogékony a játékokra, és az okos eszközöket inkább a kikapcsolódás mint a munka eszközének tekinti. Ezért kézenfekvő a gamification eszköztárával közelíteni meg a felhasználó ösztönzésének kérdését. Egy meta-játékot fogunk építeni a mérések és erőforrás-felajánlások köré, melynek állását a felhasználó az eszközén követheti, versenghet ismerőseivel és megoszthatja sikereit közösségi oldalakon. Ez egyszerre szolgálja a felhasználók megtartását, és új felhasználók elérését.

Olyan mérésekre is lehetőséget adna a rendszer, melyek a felhasználó aktív részvételét igénylik, feladatokat adnak a felhasználónak, melyeket véghez víve további pontokat, jutalmakat kaphat a felhasználó. Ez egyrészt hatékonyabbá teheti a mérést - például egy cella-lefedettségi térkép készítésénél így el lehet érni kevésbé frekventált pontokat is - másrészt egy újabb játékelemet vezet be, mely szórakoztatóvá teszi az egyébként csak a háttérben futó alkalmazást.

### Felhasználói profil

A kinyert mérési eredmények anonim módon kerülnek a kutatókhoz, miközben a felhasználók saját nevük alatt vesznek részt a platformhoz csatlakozó meta-játékban. Ez két különböző felhasználói profilt és anonimizálást tesz szükségessé.

#### Felhasználói adatok

A mérések egy része felhasználói viselkedésre is vonatkozhat. Az ilyen mérések kiértékeléséhez szükség van a felhasználó profiljának néhány részletére: származás, nem, kor, iskolai végzettség, nyelvismeret... ezekről a felhasználó nyilatkozhat.

A felhasználót triviálisan azonosító személyes adatok mellett (IMEI szám, felhasználó neve, telefonszáma, email-címe, kontaktok nevei, üzenetek tartalma...) a mérés is eredményez olyan adatokat, melyek megkönnyítik a felhasználó azonosítását. Például az éjszakai és nappali pozíció, elérhető WiFi hálózatok segítenek meghatározni a lakcímet és a munkahelyet vagy oktatási intézményt. Ugyanakkor ezek az információk a mérés szempontjából is fontosak lehetnek. Lehetőséget kell adni a felhasználónak arra hogy rendelkezzen ezen bizalmas információk fölött.

#### Mérési adatok

A mérési adatok a meta-játék során kompromittálódhatnak. Ennek megakadályozása könnyebb feladat mint a felhasználó elrejtése a kutató elől, de figyelmet igényel, hogy ne szivárogjanak ki trófeákon, érmeken, rangokon keresztül, melyek többet árulnak el a felhasználó környezetének, mint amennyit a felhasználó szeretne.

## 2. 3. Kutatók

Mint említettük, a kutatók a rendszer igazi felhasználói. A projekt elején a környezetünkben felmerülő kutatási témákhoz fogjuk szabni a DroidLab funkcionalitását, a fejlesztés alatt csak belső kutatásokat fogunk támogatni. Ahogy a hálózat bővül, úgy fogunk folyamatosan nyitni a külső kutatási projektek felé.

A rendszer fejlesztése során a kutatók kiszolgálása az utolsó lépések közt szerepel, elsődleges feladat a DroidLab mérőhálózatának kiépítése, a kutatói felületek csak az értékesítéshez szükségesek.

### Kutatói igények

#### Magas szintű igények

##### Minimális konfigurálás

A kutatók olyan rendszert szeretnének, mely elfedi előlük az adminisztrációs feladatokat. A tesztrendszerek, szimulátorok egy része ott bukik meg, hogy túl sokat vár el a kutatótól. A DroidLab-ot ezért úgy terveztük, hogy a kutatónak minimális ráhatása legyen a rendszer működésére. Nem kell konfigurációs file-okban specifikálnia, hogy milyen képességű csomópontokon, mikor milyen kód fusson, milyen topológiába szerveződjenek a csomópontok, milyen kapcsolatok épüljenek ki a csomópontok közt. A kutatónak néhány egyszerű paraméterrel kell csak megadnia általános irányelveket: hogy a mérései független vagy kapcsolódó felhasználók eszközein fussanak-e, és hogy milyen erőforrások, szolgáltatások szükségesek a mérések lefutásához. A rendszer fog gondoskodni arról hogy kiválassza a megfelelő eszközöket, és továbbítsa a méréshez szükséges információkat.

##### Könnyen tanulható nyelv

A kutató nem szeretne teljesen új programozási nyelveket tanulni csak azért hogy egy egyszerű mérést leírjon. Ezért kezdetben Java, majd a rendszer kiterjesztésével biztonsági okokból LUA nyelvű kódot fog futtatni a DroidLab.

##### Tesztelési lehetőség

A kutatónak lehetőséget kell biztosítani arra, hogy az általa készített kódot saját eszközén tesztelje, az éles rendszerbe csak olyan modult töltsön föl, ami a számára fontos méréseket végzi az általa igényelt kimeneti formátumot generálva. Ennek az igénynek a kielégítésére létre fogunk hozni egy teszt-disztribúciót, mely nem tartalmazza kvóta-ellenőrzéseket, és egy egyszerűbb, egy felhasználós szervert használ. A tesztkörnyezet egyszeri telepítésével a kutatónak lehetősége lesz több modult is vizsgálni. Ezt a környezetet fogjuk a plugin fejlesztéshez is használni.

#### Alacsony szintű igények

Az okostelefonok mélyen be vannak ágyazva a felhasználók életébe mind fizikailag mind funkcionálisan. Az Android eszközök sok fizikai szenzorral rendelkeznek a világgal való interakció megkönnyítésére és hozzáférést biztosítanak a felhasználó adataihoz és korlátozott mértékben a viselkedéséhez is. Ez rengeteg mérhető tulajdonságot eredményez.

Ennek megfelelően a kutatói igények is szerteágazóak lesznek, a fő kategóriák:

* Adatelérés: a felhasználó adatainak elemzése, például telefonkönyv hossza, tárolt file-ok mennyisége, telepített alkalmazások száma...
* Passzív mérések: a telefon állapotának lekérdezése, például pozíció, telefonhívás minták, akku töltöttség, adatforgalom, futó alkalmazások, képernyő állapota...
* Aktív mérések: olyan mérések, melyek a telefon aktív részvételét igénylik, például környező Bluetooth egységek, elérhető WiFi hálózatok felderítése, elérhető sávszélesség, RTT, csomagvesztés mérése saját adatfolyam segítségével
* Eszközök közti kommunikáció: DroidLab hálózatban részt vevő eszközök közti adatforgalmazás akár Bluetooth, akár WiFi direct vagy IP alapú csatornákon.
* Felhasználói interakció: A mérés kérheti a felhasználót valamilyen beavatkozásra, például pozíció változtatása, véleménynyilvánítás valamilyen kérdésben (szolgáltatás-minőség, aktuális tevékenység, közérzet...)

A keretrendszer fejlesztésekor nem fogjuk tudni kiszolgálni az összes különböző kutatói igényt, az eszközök fejlődésével folyamatosan szélesedik a mérhető tulajdonságok, megfigyelhető események száma. Ezért olyan rendszert terveztünk, mely könnyen bővíthető, egységesen kezeli a különböző erőforrásokat.

## 2. 4. Fejlesztők

A rendszer fejlesztésében a projekt első szakaszában csak a BME TMIT alkalmazottai vesznek részt. A gyorsan formálódó architektúra és kódbázis nem teszi hatékonnyá külső fejlesztők bevonását. Azonban már most látszik hogy a DroidLab fejlesztése több erőforrást igényel. A későbbi fejezetekben ismertetett szoftver architektúra lehetőséget ad arra, hogy egyes feladatokat külső fejlesztők végezzenek el. Elsősorban a rendszert használó kutatók lesznek érdekeltek abban, hogy a platform bizonyos komponenseit kiterjessszék saját igényeiknek megfelelően.

# Üzleti modell

A kliens-alkalmazás ingyenesen lesz elérhető a Google Play alkalmazás-boltban. Sőt, hosszú távon a rendszerbe való részvételt közvetlen - erőforrás alapú díjazás - vagy közvetett - nyereményjátékok, versenyek díjazása- támogatni is fogjuk.

A DroidLab egy open source projekt lesz, a kódbázist a GitHub-on vagy valamely más, nagy közönségbázissal rendelkező git hosting szolgáltatónál fogjuk tartani. Open source projekteknél több üzleti modell is megfigyelhető.

* Kettős licenszelés: a szoftver egy része szabadon terjeszthető, más komponensei pedig értékesítésre kerülnek. Az ingyenes szolgáltatásokat megkedvelő felhasználók egy része fog fizetni, hogy kiterjessze szolgáltatásait.
* Freemium: Eddig elsősorban játékokban használták, hasonló a kettős licenszeléshez, de itt a szoftver működéséhez igazodik a dekompozíció. A szoftverben megjelenő virtuális javakért kell fizetni, akár valamely funkció használatáért, akár olyan tartalomért mely megkönnyíti vagy élvezetesebbé teszi a szoftver használatát (egyedi skin-ek, média-tartalom...)
* Szoftver szolgáltatásként (SaS): a szoftver forráskódja szabadon hozzáférhető és futtatható, egy software hosting szolgáltatás kerül értékesítésre, a felhasználó azért fizet, hogy a szoftvert a kiadó infrastruktúráján futtattassa, megkímélve magát a hardware beszerzéstől és rendszer-adminisztrációtól.
* Nem szabadon felhasználható szoftver: a nyílt forráskódú szoftver továbbértékesítése licenszköteles.
* Technikai támogatás értékesítése: a szoftver szabadon használható, a kliensek támogatási csomagokért fizetnek, melyek az egyszerű felhasználói támogatáson túl tartalmazhatnak testre szabást, rendszertervezést és telepítést

A DroidLab modellje leginkább a SaS modellhez hasonlít. A DroidLab a felhasználók eszközeiből tevődik össze, ezért a kutatók hiába rendelkeznek a DroidLab forráskódjával, a használathoz ki kellene építeniük egy megfelelő méretű felhasználó-bázist. Ez idő vagy pénzigényes, és egy piacvezető alkalmazás mellett nehezen kivitelezhető. Ezért a kutatók rá vannak utalva arra a hivatalos DroidLab alkalmazás felhasználóbázisára. Fizetni a későbbiekben bemutatott kvóta-rendszer alapján fognak a felhasználói erőforrásokért.

A mérések hostolása mellett terméktámogatást is érétkesíteni fogunk: a programozásban nem jártas kutatóknak segítünk majd tesztjeik, méréseik implementálásában. Ha a kutatáshoz speciális igények merülnek fel, melyet a keretrendszer eddig nem támogatott, a pluginek kiterjesztése is értékesíthető lesz, hiszen hiába készítené el a kutató az elérhető kódbázis alapján saját kiegészítését, az csak akkor kerül fel a felhasználói eszközökre, ha a hivatalos alkalmazást frissítjük vele.

# A Rendszer felépítése

## 4. 1. Kliens

A felhasználók Android alapú készülékein futó szoftvert tekintjük kliensnek. A kliens szinte minden készüléken futtatható, két elvárást támasztunk az eszközökkel szemben: rendelkezzenek hálózati hozzáféréssel, és érjék el a Google szolgáltatásait. Ezek nem erős megkötések, minden felhasználó érdeke a Google alkalmazások telepítése, és valamilyen hálózati kapcsolattal minden felhasználói eszköz rendelkezik.

A munkánk első szakaszában elkészítettünk egy kliens-prototípust, melyen felhasználói és fejlesztői oldalról is vizsgálható elképzeléseink helyessége. A következőkben a prototípusból levont tapasztalatokat és a kialakuló szoftverarchitektúrát és annak működését mutatjuk be.

### Pluginek

Az Android platform jogosultságkezeléséről már esett szó korábban. Ez egy erős érv volt a kliens modulárissá tételére. Egyrészt a felhasználó nehezebben bízik meg egy olyan alkalmazásban, mely minden jogot elkér telepítésekor. Másrészt az alkalmazás funkcionalitásának kiterjesztésével minden frissítéssel nőne a szükséges jogok száma.

Egy monolitikus rendszerben az egyes funkciók bővítése újabb és újabb frissítések letöltését tenné szükségessé. E mellett sok olyan komponens felkerülne a felhasználó telefonjára, amit esetleg hardware-es korlátok miatt ki sem tud használni a mérőrendszer.

Ezért a prototípusban a következő felépítés mellett döntöttünk: az egyes szolgáltatás-csoportokat pluginekként implementáljuk. Minden plugin egy önálló Android alkalmazás lesz. Az Android keretrendszer megfelelő támogatást biztosít ezen alkalmazások authentikálására így a keretrendszernek lehetősége van ellenőrizni a telepített plugineket és biztonságos csatornán kommunikálni velük. Le tudja kérdezni a pluginek verziószámát, és lehetőség van egy plugin letöltését vagy frissítését kezdeményezni.

A mérőrendszer funkcionalitását a pluginek adják, becsomagolják az eszköz által biztosított szolgáltatásokat. A DroidLab fejlesztése során két plugin készül: a Bluetooth és a Cellular pluginek lefedik az összes lehetséges plugin-használati módot így alkalmasak a koncepcionális hibák feltárására és javítására. A további plugineket a kutatók igényeihez igazodva fogjuk ütemezni és fejleszteni, vagy bevonni a kutatókat a fejlesztésbe. Előreláthatólag a következő plugineket terveztük:

* Bluetooth
* Cellular
* WiFi
* Adatforgalom
* Lokáció
* Szenzorok
* NFC
* Telefónia
* Alkalmazások, felhasználói viselkedés
* Kontaktok, közösségi hálózatok
* Processzor és memória-használat, akkumulátor, üzemidő

De ez a lista az eszközök fejlődésével tovább bővülhet. Az egyes pluginek funkcionalitása még nem körvonalazódott, ezeket a döntéseket akkor fogjuk meghozni amikor megjelennek a kutatói igények.

Az Android 4.0-ás verziójától kezdve minden telepített alkalmazásnak rendelkeznie kell futtatható Activity-vel, így a pluginek is meg fognak jelenni az alkalmazás-listában. Ez egyfelől kellemetlen, mert így a DroidLab akár 6-8 ikonnal is hozzájárul az alkalmazáslistához. Ám ezt a lehetőséget arra is ki lehet használni, hogy közelebb hozzuk a felhasználót a keretrendszerhez, betekintést engedve neki a rendszer működésébe. Egy egységes arculat mögött minden plugin bemutatkozhat, beállítási lehetőségeket kínálhat a felhasználó felé, és a plugin működésével kapcsolatos lokális és összesített statisztikákkal buzdíthatja a felhasználót további erőforrások megosztására.

### Modulok

A kutatók által fejlesztett méréseket hívjuk moduloknak. A modulok a prototípusban Java nyelven készülnek, ezt biztonsági okokból a későbbiekben egy magasabb szintű script nyelv, valószínűleg a LUA, de lehet hogy egy saját fejlesztésű nyelv fogja felváltani.

A modulok olyan futtatható kódrészletek kell legyenek, melyek felett a keretrendszernek teljes ellenőrzése van. Ezért a modulok nem kommunikálhatnak közvetlenül az Android platformmal.Nem indíthatnak saját szálat, nem iratkozhatnak fel rendszer-eseményekre. Csak így biztosítható a felhasználó védelme.

A biztonsági követelményeknek a Java kód nem felel meg. Különböző trükkökkel tetszőleges kódrészlet futtatható lenne, többek közt olyan is, mely magát a keretrendszert vezérelné így teljes ellenőrzést gyakorolna a felhasználó eszköze fölött. Ezért a moduloknak biztonsági konténerben kell futniuk.

A DroidLab nem egy elosztott párhuzamos szuperszámítógép. A modulok a pluginek funkcióit felhasználva végeznek méréseket. Ezek a mérések vagy periodikusak, vagy eseményvezéreltek, a mérendő tulajdonság vagy jelenség jellegéhez igazodva. A kétféle viselkedés a modul kódjában elkülönül. Elnevezési konvenciókkal biztosítjuk a kétféle működés megvalósítását.

A modulok számára a keretrendszer perzisztens tárat biztosít a részeredmények tárolására, és egy loggolási lehetőséget az eredmények rögzítésére.

### Keretrendszer

A DroidLab alapfunkcionalitását megvalósító alkalmazást nevezzük keretrendszernek. A keretrendszer feladata a telepített pluginek megkeresése és igény szerinti frissítése vagy letöltése. A keretrendszer tölti le, indítja el és állítja le a modulokat, tölti föl a mérési eredményeket.

A prototípusban megvalósítottuk a pluginekkel és modulokkal való kommunikációt. A pluginek felé a különböző kipróbált megoldások közül végül a címzett Broadcast Intentekre esett a választás, ezek mind teljesítményben, mint biztonságban megfelelnek a követelményeknek. A pluginek egységes kezeléséről egy általunk kidolgozott szöveg alapú protokoll gondoskodik.

A modulok jelenleg Java nyelven készíthetőek, dex formátumú bináris állományokként kerülnek az eszközre ahol betöltődnek és példányosításra majd meghívásra kerülnek a megfelelő metódusok. Hogy a rendszer ne fogyasszon indokolatlanul sok erőforrást, a későbbi script alapú megvalósításokat a szerver fogja Java-ra fordítani és csomagolni, és a készülékek a későbbiekben is a prototípusban implementált megoldást használhatják.

### Kvóták és jogosultságok

A pluginek a teljes funkcionalitásukhoz szükséges jogosultságokat el kell kérjék településükkor. A felhasználónak azonban lehetősége van ezeket a jogosultságokat korlátozni a plugin beállítási felületén. A plugin fejlesztőinek a felelőssége a megfelelő granularitású jogosultságokat kidolgozni, hogy a felhasználó tudatos döntést tudjon hozni a plugin funkcionalitásának korlátozásáról.

A johosultságok mellett a pluginek kvótákat is definiálnak. A kvóták testesítik meg a felhasználó által felajánlott erőforrásokat. A kvóták is a pluginekre jellemzőek, lesznek pluginek, melyekre egyáltalán nem vonatkozik kvóta, lesznek, melyek többféle kvótát is definiálnak. A DroidLab rendelkezésére bocsájtott erőforrások mértékét a felhasználó állapítja meg. A felhasználó által beállított értékeket a keretrendszer sosem fogja túllépni, ezért is fontos a menedzselt modulkód.

A modulok is rendelkeznek kvótákkal, ezeket azonban nem a felhasználó szabályozza. A központi vezérlő feladata, hogy a felhasználók által felajánlott kvótákhoz modulokat rendeljen. A kutatók meghatározzák, hogy hány készüléken, milyen mennyiségű erőforrásra van szükségük, a vezérlő ezeket az igényeket párosítja a felajánlásokkal, majd küldi a modulokat a megfelelő eszközökre. Az eszközök a modul kódja és leírása mellett a modulhoz rendelt erőforrás-mennyiséget is megkapják, a keretrendszer nem csak a teljes kvóta-limitet, hanem a modulokhoz rendelt kvótákat is betartja. Minden mérés addig fut, ameddig a határideje le nem jár, vagy ki nem fogy valamelyik igényelt kvótából. Ekkor a mérés lezáródik.

### Felhasználói felület

Az alkalmazás alapfunkcionalitásához nincs szükség felhasználói interakcióra . Ezért a felületet nem az alapfunkcionalitás, hanem a hozzáadott funkciók ihlették.

A felhasználói felület a DroidLab felépítéséhez igazodva moduláris. Az alkalmazás főképernyőjéről érhetőek el az egyes modulok és pluginek képernyői. A felhasználót nem csak tájékoztatjuk az eszközön futó mérésekről, hanem döntési lehetőséget is adunk neki.

#### Keretrendszer felülete

A keretrendszer felülete általános híreket és statisztikákat közöl a rendszerről. Ezen a felületen tudnak a hálózat üzemeltetői kommunikálni a felhasználóval, felhívások, hibaüzenetek és egyéb visszajelzések közvetíthetőek a felhasználók felé.

#### Plugin felületek

A felhasználók jelentős része nem rendelkezik a készülékének kezeléséhez szükséges ismeretekkel sem, ezért nem szabad nehéz döntések elé állítani, de közben rá kell bízni a jogosultságok az erőforrások kiosztását.

A jogosultságoknál minden egyes jogosultságnál példákat adunk a rendeltetésszerű használatra, és felhívjuk a figyelmet az esetleges hátrányokra, visszaélési lehetőségekre. Például egy SMS-ekhez kötődő jog lehet a következő: "SMS statisztikák: Az SMS-ek érkezési idejének, olvasási és válaszadási idejének rögzítésével a kutatók képet kaphatnak a felhasználó tipikus SMS kommunikációs szokásairól. A jog nem ruházza fel a kutatókat az SMS tartalmának megismerésére. Az SMS feladóját anonimizált formában biztosítjuk a kutatók számára."

A jogosultságok által hozzáférhetővé tett erőforrásokra vonatkozó kvóták a jogosultságok mellett, azok megadás esetén jelennek meg a felületen.Hogy ne kelljen olyan döntéseket hoznia, hogy 10 vagy 200 Bluetooth felderítést engedélyez naponta az eszköznek, a pluginek fejlesztése során néhány tipikus értéket választunk ki minden erőforráshoz. Az erőforrás jellegéből adódóan 3-10 különböző, jól leírt erőforrás-mennyiséget határozunk meg. Egy adatforgalomra vonatkozó kvóta például a következő értékekkel kerül bemutatásra: Csak szöveges dokumentumok, 1-2 weboldal, néhány zeneszám, egy teljes album, egy film egy DVD lemez, egy BluRay lemez, mely értékek a 100kB, 1.5MB, 10MB, 100MB, 700MB, 4.5GB, 25GB volumenű adatforgalomnak felelnek meg. Így a felhasználók érezni fogják a kvóták nagyságrendjét és tudatos döntést tudnak hozni.

#### Modul-felületek

A modul lista egyik célja a felhasználó motiválása a rendszerben való részvételre. Ha a felhasználó számára érdekes, vagy általa a társadalom számára hasznosnak talált projektekben vesz részt, akkor a márásekhez is szívesebben járul hozzá. A modulok felülete részletes információt nyújt a modul által igényelt és elhasznált erőforrásokról, a futás esetleges részeredményeiről, és magáról a mérésről.

A másik vél a felhasználó eszköze felett gyakorolt felügyeletének megerősítése. A felhasználónak lehetősége van bármelyik ,számára kártékonynak vagy haszontalannak tűnő modul eltávolítására.

### Mérési adatok kezelése

A keretrendszerrel nem szerettük volna szabályozni a modulok kimeneti formátumát. Így egy, a modul kimenetét leíró meta-nyelv helyett az egyszerűbb utat választottuk. A modulok kimenete szabad szöveg lesz. Ezek a sorok időbélyeggel és a futtató eszköz azonosítójával ellátva kerülnek tárolásra a kliensen, amíg megfelelő lehetőség nem adódik a szerverre való feltöltésükre.

A modulok mellett a keretrendszer is küld a szervernek mérési adatokat. Minden plugin definiálhat generikus eseményeket és generikus adatokat, amikre a keretrendszer feliratkozik és amiket a keretrendszer polloz. Ezeket az általánosan felhasználható adatokat a szerveren később minden kutató számára elérhetővé tesszük. Ez csökkenti a rendszer terhelését, hiszen azok a kutatók, akiknek csak ezekre az adatokra van szükségük, nem kell hogy méréseket futtassanak értük, historikusan a teljes rendszer történetére és az összes felhasználóra elérik azokat. A kutatók által készített többi adat csak azok tulajdonosai által érhető el, logfile-ok formájában. A DroidLab az adatok elemzéséhez nem nyújt támogatást.

## 4. 2. Szerver

A szerver a munka első szakaszában csak támogató tevékenységet végzett, fogadta a prototípusról érkező mérési adatokat, rögzítette a készülékek státuszát, és néhány modult elérhetővé tett a telefonok számára. Ezeket a funkciókat egy PHP alapú website valósította meg.

A szerver felületére a későbbiekben a kutatók léphetnek majd be, itt fogják feltölteni a modulok kódját és a konfigurációs file-t, ami a modul igényeit és rövid bemutatását tartalmazza, és itt érhetik majd el a mérésük státuszinformációit és a mérés végeztével a generált logokat.

A szerver legfontosabb feladata a kliensek nyomon követése lesz. A kliensek felajánlott erőforrásaikról jelentést küldenek a szervernek, ami ezek alapján ütemezi és osztja ki a modulokat az eszközök közt. A fentieken túl a szerver kell támogassa a gamification-t, és elérhetővé kell tennie a kliensek számára a rendszer működését leíró számokat érdekes információkat, hogy a keretrendszer felülete színesebb legyen.

A komplex funkcionalitás és a jó skálázhatóság érdekében a PHP-ről hosszú távon át fogunk térni egy felhő alapú megoldásra, valószínüleg a Google Appengine technológiát fogjuk használni.

# Konklúzió

A munka első szakaszának eredménye egy prototípus-alkalmazás volt, mely megvalósította a végleges alkalmazás legfontosabb funkcióit. A kliensben ez a pluginek és modulok kommunikációja, melyre több technikai megoldást is kidolgoztunk és kimértünk, mielőtt a végleges kiválasztásra került volna. A szervernek csak az alapfunkcióit készítettük el, elsősorban a kliensalkalmazás támogatására, az ütemező és a kutatói felületek a projekt utolsó szakaszába kerülnek. E mellett megvizsgáltuk a rendszert a különböző felhasználók szemszögéből, elemeztük a lehetséges scenariokat és üzleti modelleket.

A projekt következő szakaszában a kliens véglegesítésére fogunk koncentrálni, elkészítjük a biztonságos modul-konténert és az erőforráskvóták követését.

Összességében az eredeti vállalások kivitelezhetőnek tűnnek, eddig nem derült fény olyan limitációra mely veszélyeztetné a megvalósíthatóságot. Valószínűleg az ütemezési algoritmus skálázhatósága fog majd problémát okozni, itt egy szuboptimális megoldással kell majd kiegyezni. A gamification és felhasználó motiválás, és az elérhető erőforrások kutatók közti szétosztása pedig túlmutat már a projekt jelenlegi keretein.

Tartalom

[1. Bevezetés 2](#_Toc354642301)

[2. A Droidlab környezete 2](#_Toc354642302)

[2. 1. Eszközök 2](#_Toc354642303)

[Felhasználói eszközök 3](#_Toc354642304)

[Központi vezérlő 3](#_Toc354642305)

[2. 2. Felhasználók 3](#_Toc354642306)

[Felhasználói élmény 4](#_Toc354642307)

[Felhasználói profil 5](#_Toc354642308)

[2. 3. Kutatók 6](#_Toc354642309)

[Kutatói igények 6](#_Toc354642310)

[2. 4. Fejlesztők 7](#_Toc354642311)

[3. Üzleti modell 8](#_Toc354642312)

[4. A Rendszer felépítése 9](#_Toc354642313)

[4. 1. Kliens 9](#_Toc354642314)

[Pluginek 9](#_Toc354642315)

[Modulok 10](#_Toc354642316)

[Keretrendszer 10](#_Toc354642317)

[Kvóták és jogosultságok 11](#_Toc354642318)

[Felhasználói felület 11](#_Toc354642319)

[Mérési adatok kezelése 12](#_Toc354642320)

[4. 2. Szerver 13](#_Toc354642321)

[5. Konklúzió 13](#_Toc354642322)