

H2020-ICT-688712



Project: H2020-ICT-688712

Project Name:

5G Applications and Devices Benchmarking (TRIANGLE)

Deliverable D5.2

Report on App Lab Integration with FIRE testbed

Date of delivery:	01/09/2016	Version:	1.0
Start date of Project:	01/01/2016	Duration:	36 months



Deliverable D5.2

Report on App Lab Integration with FIRE testbed

Project Number:	ICT-688712
Project Name:	5G Applications and Devices Benchmarking
Project Acronym	TRIANGLE

Document Number:	ICT-688712-TRIANGLE/D5.2
Document Title:	Report on App Lab Integration with FIRE testbed
Lead beneficiary:	University College London
Editor(s):	Niall Roche (UCL)
Authors:	Niall Roche (UMA),
Dissemination Level:	PU
Contractual Date of Delivery:	31/08/2016
Work Package Leader:	Niall Roche (UCL)
Status:	Final
Version:	1.0
File Name:	TRIANGLE_Deliverable_5.2-final.docx

Abstract

App Lab platform is currently in production at UCL to support real user based testing of mobile applications. PerformNetworks is a FIRE testbed deployed at UMA for experimentation in 4G/5G technologies. TRIANGLE is a new FIRE testbed coming from the extension and combination of both platforms in the context of the project. This deliverable discusses the potential approaches for the integration and present the final decision to be implemented.

Keywords

Testing Mobile apps; Testing Portal; Testbed integration



Document: ICT-688712-TRIANGLE/D5.2

Date: 01/09/2016

Dissemination: PU

Status: Final

Version: 1.0

Executive summary

App Lab is a private app store which functions as an app distribution platform for a user community on UCL campus. This allows students and staff to test early versions of software developed by SMEs pre-release on public App Stores. For the TRIANGLE project App Lab plays a part in the testing of apps on humans to determine their feedback on the app and their perceived Quality of Experience in using the app under different network conditions. Physical integration between App Lab and the TRIANGLE Testbed components was discussed during the project proposal stage, but on discussion it was felt to be unnecessary as any testing would not be possible as not all elements of the network path would be controllable and latency would invalidate any benefit from doing this.

A functional integration and lightweight interconnection for exchanging uploaded mobile applications over a VPN connection will however be implemented. This approach will make possible to take advantages of both platforms, App Lab at UCL and the FIRE testbed at UMA. Future potential integrations with TRIANGLE Testbed elements have been identified when UCL's testing lab is at a further stage of development later in the project.



Contents

1	Introduction	1
2	App Lab Overview	2
2.1	App Lab Background.....	2
2.2	User Testing and feedback through App Lab	2
2.3	Audio and Video application trialing.....	3
2.4	App Lab Architecture.....	3
2.5	App Lab Portal	4
3	App Lab TRIANGLE Testbed Integration Overview	6
3.1	App Lab as a App Submission and TRIANGLE Portal for Triangle	6
3.2	VPS integration with App Lab.....	6
3.3	VPN integration with App Lab.....	6
3.4	Remote App Testing on devices.....	7
3.5	Other uses of App Lab	7
3.6	Deployment of App Lab University Malaga.....	7
3.7	Planned use of App Lab	7
3.7.1	App Lab using the Triangle Testbed for device testing and validation	8
3.8	TRIANGLE Portal integration with App Lab	9
4	App Lab TRIANGLE Testbed Integration Summary	11
5	References.....	12
6	Annex 1. App Lab iOS Screenshots	13
7	Annex 2. App Lab Portal Screenshots.....	16
8	Annex 3. App Lab Software Architecture	19
9	Annex 4. VELOX VPS Engine	20



Document: ICT-688712-TRIANGLE/D5.2

Date: 01/09/2016

Dissemination: PU

Status: Final

Version: 1.0

List of Figures

Figure 1 - App Lab High- Level Architecture	4
Figure 2 - App Lab App Submission Process.....	5
Figure 3 - App Lab Triangle Flow.....	9



Document: ICT-688712-TRIANGLE/D5.2

Date: 01/09/2016

Dissemination: PU

Status: Final

Version: 1.0

List of Tables



List of Abbreviations

API	Application Programming Interface
E2E	End-to-End
JSON	JavaScript Object Notation
JEE	Java Enterprise Edition
LTE	Long Term Evolution
Mbps	megabits per second
Mo	Month
MA	Multiple Access
MAC	Medium-access Control
QoE	Quality of Experience
QoS	Quality of Service
SLA	Service Agreement Level
SME	Small and medium sized enterprises

TCP	Transmission Control Protocol
UCL	University College London
UE	User Equipment
UL	Uplink
UMA	University of Malaga
UUID	Universally Unique Identifier
UX	User eXperience
VPS	Virtual Path Slice
VPN	Virtual Private Network
WLAN	Wireless Local Area Network
WP	Work Package
WPAN	Wireless Personal Area Networks



Document: ICT-688712-TRIANGLE/D5.2

Date: 01/09/2016

Dissemination: PU

Status: Final

Version: 1.0

1 Introduction

This document provides an overview of App Lab and its relationship to the TRIANGLE Project, in particular how it fits into the overall certification process and the Testbed and its components. The consortium had a number of conversations and ideas about how App Lab can benefit the project, and where it could fit architecturally with the rest of the Testbed Components, both at present, and at later stages of the project. This document will outline the various ideas that were proposed and may be considered at a later stage in the project and plans for the current integration between the functionality of App Lab and the TRIANGLE Testbed.



2 App Lab Overview

This section contains an overview of App Lab [1], what is used for and the benefits that it provides and how it will benefit the TRIANGLE Project and TRIANGLE Testbed application users. Some screenshots of the App Lab web admin portal and the iOS Client application are included in Annex 1.

2.1 App Lab Background

App Lab is a mobile app distribution platform developed by UCL. App Lab has been developed by UCL Advances[2] (now renamed to Innovation and Enterprise), UCL's center for entrepreneurship, which was formed to develop new links between UCL and businesses of all kinds, with a particular focus on entrepreneurs and small and medium sized enterprises (SMEs). UCL Advances ran a number of programmes to deliver a unique, in-depth service to help SMEs to test new products and services, gain customer feedback and ensure technical feasibility.

App Lab was one of these projects to create a "living- lab" testbed that provides controlled conditions for SMEs to test and validate their innovations, through access to UCL's 45,000 staff and students. This project supports innovation and early product development in a mobile ecosystem environment.

App Lab was developed as an app testbed and is a high level platform for distributing applications to a large- scale testbed for pre-commercial testing and validation services. To accomplish this, App Lab provides a private (enterprise) app store, offering in-the-wild user rapid field testing and feedback.

The system is independent of the Apple iOS App Store[3] and Google Play[4], limited to authorized users of the University community, under the Apple Enterprise Agreement for iOS[5], which for universities allows distribution to its students and staff under limited conditions. Google do not provide restrictions for their App Distribution.

2.2 User Testing and feedback through App Lab

In addition to app distribution, App Lab collects valuable data for evaluation to provide fast iteration of releases for app improvement cycles. The data is useful to see how users are engaging with the applications, what features are used and not used, and how user journeys occur with real users. App Developers can use this data to tweak elements such as the User interface, or to try different experiments such as A/B Testing [7] to see if changes have an effect on users in a subsequent release.

App Lab uses a third party tool called UXCam [6] to capture data about how users interact with the applications. UCL is negotiating an agreement with UXCam to allow free usage for the consortium members and the experimenters that will participate in Triangle through the Open Call mechanism.

UXCam allows visualization of hotspots where users are interacting with the UI and entire user journeys can be played back as a set of animations from a web console. This is very useful for SMEs to visualize how real users are interacting with the applications, as opposed to how they think users should interact with their applications. This often leads to discovering usability issues and necessitating rework of the User Experience.

In the context of Triangle UXCam can be used as a data source for extracting user behavior models or single specific behavior for scripting the App testing tools available in the testbed.



UCL are also in discussion with other analytics tracking providers, one of whom, TestFairy [9], may also be useful for tracking purposes. One of the important aspects of the data captured is the meta data capture and looking at how this data can be used to represent user journeys for the purposes of automating common or problematic user journeys for testing and validation purposes.

2.3 Audio and Video application trialing

App Lab collects valuable data for evaluation fast iteration of releases for app improvement cycles and includes audio and video capability as part of the government sponsored Innovate UK Digital Testbed. UCL have signed an agreement with Innovate UK [9] to use their Digital Testbed trial management software solution to facilitate trials of multimedia content on the student body, to determine commercial viability and user feedback for different content and delivery and charging models.

UCL and Innovate UK believe this Digital Testbed facility will allow Multimedia Content Providers to bring high value content for use by the test community and provide a strong justification for 5G technologies for high bandwidth and potentially use of the content in non-traditional delivery methods enabled via 5G, such as Augmented and Virtual Reality. This could be a great asset for some of the SMEs that will use the Triangle Testbed, in particular as they try to find out how to leverage the opportunities to leverage the new technologies and presentation methods in a way that is appealing to users.

2.4 App Lab Architecture

App Lab is comprised of client applications for Android and iOS mobile devices and a JEE server portal for application upload, management and distribution. The platform has a Web Management Console that carries out management tasks of the store including:

- upload of new applications or updates (applications developed by the company, public applications or purchased applications),
- Adding meta data such as app descriptions and screenshots
- test case definitions for testing, approval and publication of applications which will can be adapted for the TRIANGLE pilot cases
- workflow of app approval lifecycle including reviewer and tester comments
- managing multiple app versions as new releases are live and in testing
- targeting app variants for specific device operating system versions, useful when testing apps for beta releases (such as iOS 10 at the time of writing).
- Client software for the disabling of apps should there be an issue with the app or app content



2.5 App Lab Portal

The App Lab Portal supports the lifecycle from app submission, review & testing and publishing. Figure 1 below shows the flow for submitting apps into App Lab. Screenshots of the App Lab Portal can be seen in Annex 2. Initially there was some thoughts given by the TRIANGLE Consortium to adapt the App Lab Portal for use by the end user TRIANGLE App and Device Submission Portal, after some discussion and evaluation it was felt that this was probably not the best course of action as explained further in Section 3.1.

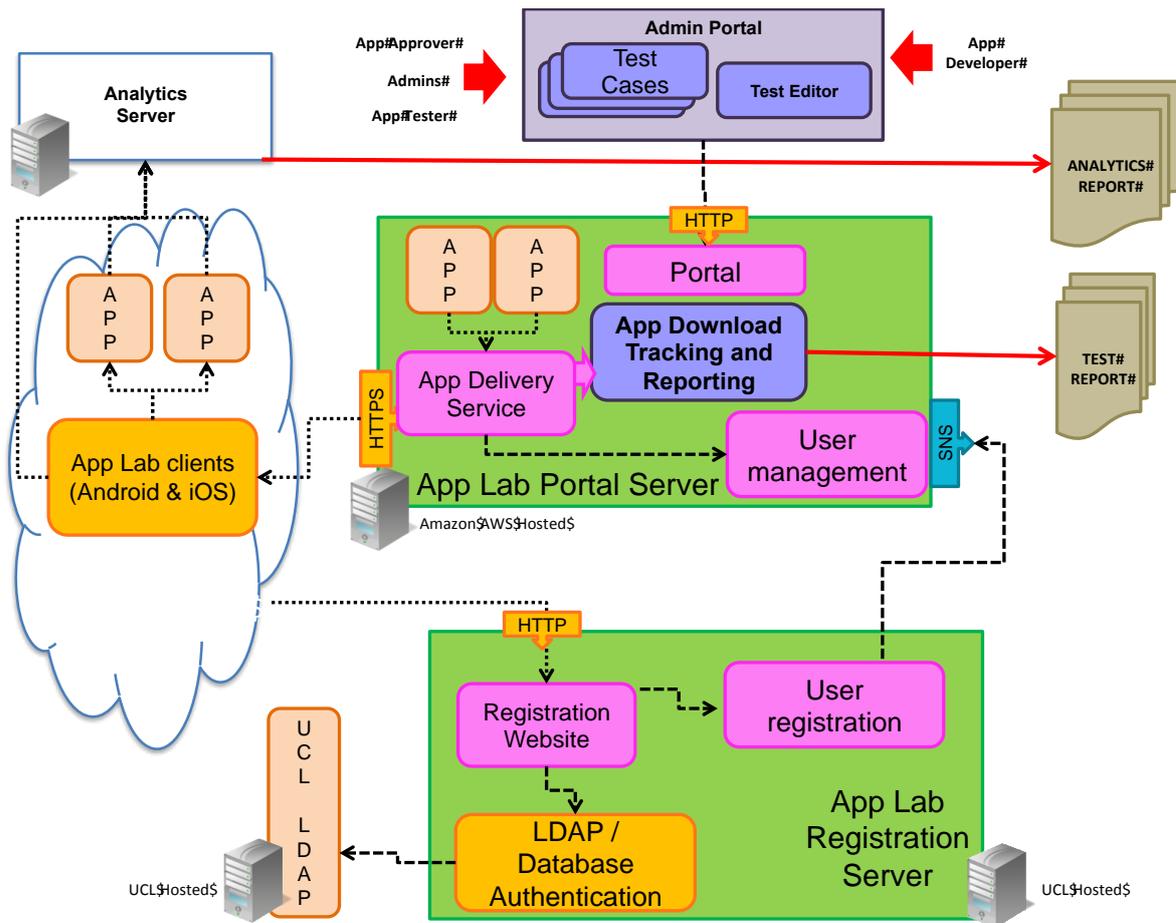


Figure 1 - App Lab High- Level Architecture

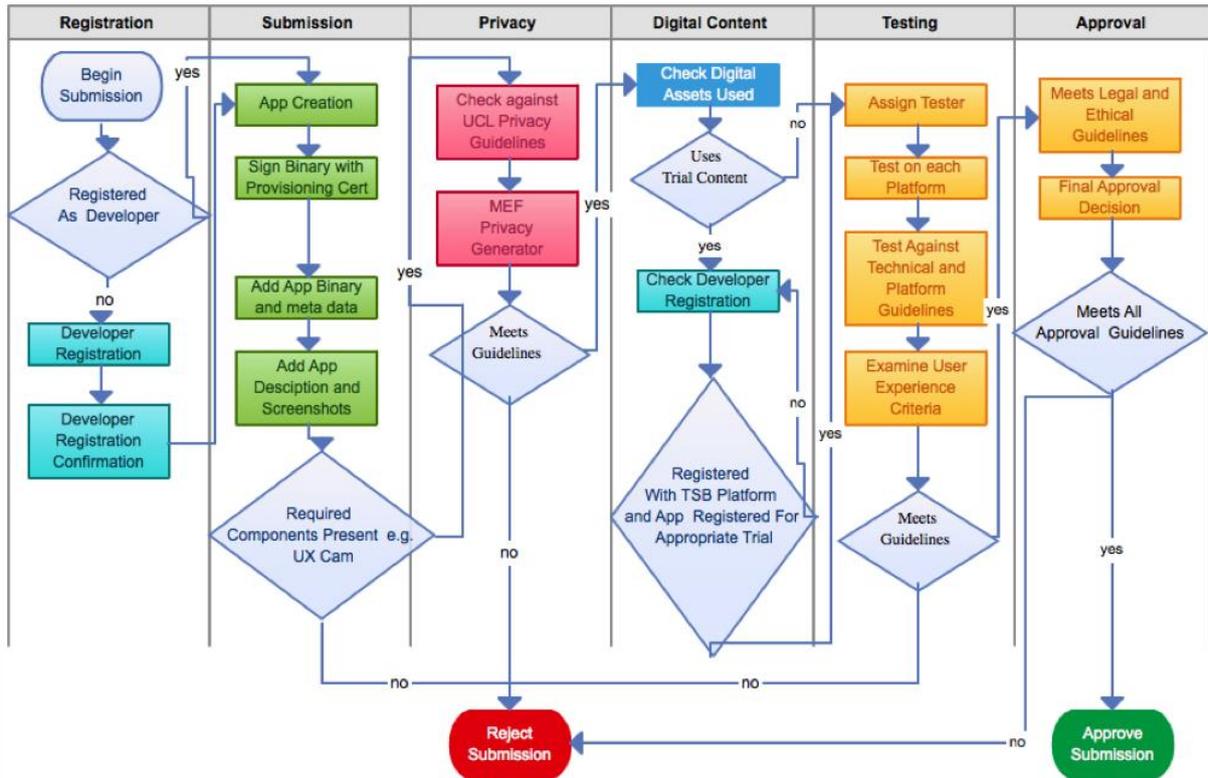


Figure 2 - App Lab App Submission Process



3 App Lab TRIANGLE Testbed Integration Overview

One of the original plans to use App Lab as part of the TRIANGLE Testbed was to create an installation locally in the Testbed based at UMA. After numerous discussions it was felt by the consortium that this would not provide much benefit to the Testbed, as the process to use both TRIANGLE and App Lab testbeds with the same app, but no physical integration is expected.

3.1 App Lab as a App Submission and TRIANGLE Portal for Triangle

One of the early ideas for a potential integration was to utilize many of the existing function of App Lab around managing app submission, upload testing distribution and approval. These high level functions had some perceived overlap with the main TRIANGLE Portal, but on more detailed discussion it was apparent that there were a number of differences between the aims and requirements of the two portals. Also the App Lab portal while quite flexible in regards to creating new kinds of tests to run as part of an app review and approval process, is not very flexible for workflow changes, though this is possible if needed.

At the London meeting in May 2016 It was decided by the consortium to proceed with LabWiki12 as a basis for the main Triangle Portal as it was flexible and would be needed to communicate with other parts of the Triangle components anyway, so it made more sense to just extend LabWiki, rather than involve an additional system.

3.2 VPS integration with App Lab

This was an early consideration and still one that we may explore later in the project. The idea was that individual apps deployed on App Lab would have a dedicated virtual path on the network with a SLA for certain bandwidth and network performance capabilities. This would use the VPS engine component from RedZinc, integrated with App Lab. In practice however, deploying an App in an environment where there is no real control over the underlying network as users in the UCL network would connect using commercial operators and campus-wide Wi-Fi networks, therefore implementing any level of path slicing is practically impossible as these networks would not be capable of being configured.

In the medium term UCL are working on creating a controllable network environment for Wi-Fi and LTE environments where it may be possible to have some type of path slice implementation. Once that is in place it may be reasonable to revisit the VPS integration.

Details of the API which applications can use to integrate on the UMA Testbed and request quality of service are given in Annex 4.

3.3 VPN integration with App Lab

Another option considered was a VPN connection between the App Lab installation at UCL and the Malaga Testbed. This was considered for a number of reasons including:

Enabling interconnection between apps distributed from App Lab and running in London and the Testbed in UMA. As mentioned in the previous section this was not practical due to the latency involved and the lack of control of the underlying network.



Document: ICT-688712-TRIANGLE/D5.2

Date: 01/09/2016

Dissemination: PU

Status: Final

Version: 1.0

There will still be a VPN connection as App Lab and the UMA Testbed components for the purposes of transferring uploaded apps from App Lab and into the Testbed and vice versa. This process and the reason for needing this is described in Section 3.7.1

3.4 Remote App Testing on devices

One idea that was explored was investigating whether it would be useful for testing Apps that are running remotely on devices in the UMA testbed, from external devices in London or elsewhere. This is technically possible but in practice of limited use. It may be useful to see what an app looks like on another device, from the form factor of a mobile device, as opposed to a mobile browser as is the usual case. However, to get an accurate feel of how the app performs and a realistic experience of response time, it is not of use due to the overhead and latency involved, however this remains a future option if a suitable use case is identified for doing it. One potential use is recording the user actions when the app is interacting with components only deployed at UMA location.

3.5 Other uses of App Lab

App Lab may still prove to be useful for the Deployment of apps inside the UMA Testbed as it is a distribution mechanism and performs features such as enterprise resigning that are likely to be useful in deploying apps in an automated fashion to the Testbed devices. There are additional components that can assist from QuaMotion12 in how iOS apps are deployed for automated testing. It is expected that this will be discussed in more detail in Month 10 at the next Triangle face to face meeting, so the best combination of iOS deployment technologies can be determined based on needs.

3.6 Deployment of App Lab University Malaga

One option that was considered would be a deployment of App Lab inside the University of Malaga with the student population. However, it was decided that an App Lab deployment would be of limited value without a sizeable user Testbed population. This would normally involve a large promotion activity to engage the student and staff bodies to download App Lab and also integration with UMA authentication mechanisms. For these reasons there are no plans to install App Lab at UMA presently.

3.7 Planned use of App Lab

At the kick off meeting in London in May 2016 it was decided that the main value of App Lab to the TRIANGLE project was in the user community and how real users actually use applications and their perceptions on how the apps performed. Real users are also beneficial where automated testing is not practical, so where automation is not possible certain tasks could be performed on apps by real users. As it was not practical to measure network behavior on apps running outside of the UCL testbed in London, there was still value in capturing information of the steps the user took inside the running application.

As a result, one benefit of App Lab is to use the information recorded by UXCam on how users have used the app and to gather that data and to use it as a mechanism to create automated scripts that will execute on Devices in the Triangle Testbed in UMA. This is important as sometimes app developers will not already have automated test scripts created



Document: ICT-688712-TRIANGLE/D5.2

Date: 01/09/2016

Dissemination: PU

Status: Final

Version: 1.0

when they submit apps to Triangle for certification, or they may have automated scripts that may not be compatible with the QuaMotion system or the popular WebDriver protocol.

In this case the data captured from UXCam could be used to generate test scripts capable of being transformed into scripts that are capable of being executed in the QuaMotion automation platform. Even where automated tests exist, these normally test the expected path through an app. In practice, these flows often differ from how real users interact with apps and also experience realistic delays in between steps as user's pause to look at screen options and decide what action to take next. One planned task is to use the recorded data from UXCam to build realistic models of the user interaction in order to automatically generate the scripts.

3.7.1 App Lab using the Triangle Testbed for device testing and validation

One immediate use of the Testbed and App Lab is to perform device and functional testing in a number of characteristics of apps that are submitted to App Lab. In the process flow illustrated below in Figure 3 we can see that users who submit their apps to App Lab for use on the UCL user community can utilise the TRIANGLE Testbed for verifying that the app functions as expected in different network and device configurations. The App Lab and Triangle Portal Interconnect will be via an FTP interface secured using VPN as outlined in Section 3.8.

This is a good pilot of the Benchmarking as a service BaaS concept that has been outlined in the Triangle Deliverable '5G BaaS – Benchmark as a Service'. As other systems may want to integrate the Triangle Testbed as part of their testing and validation or possibly certification processes.

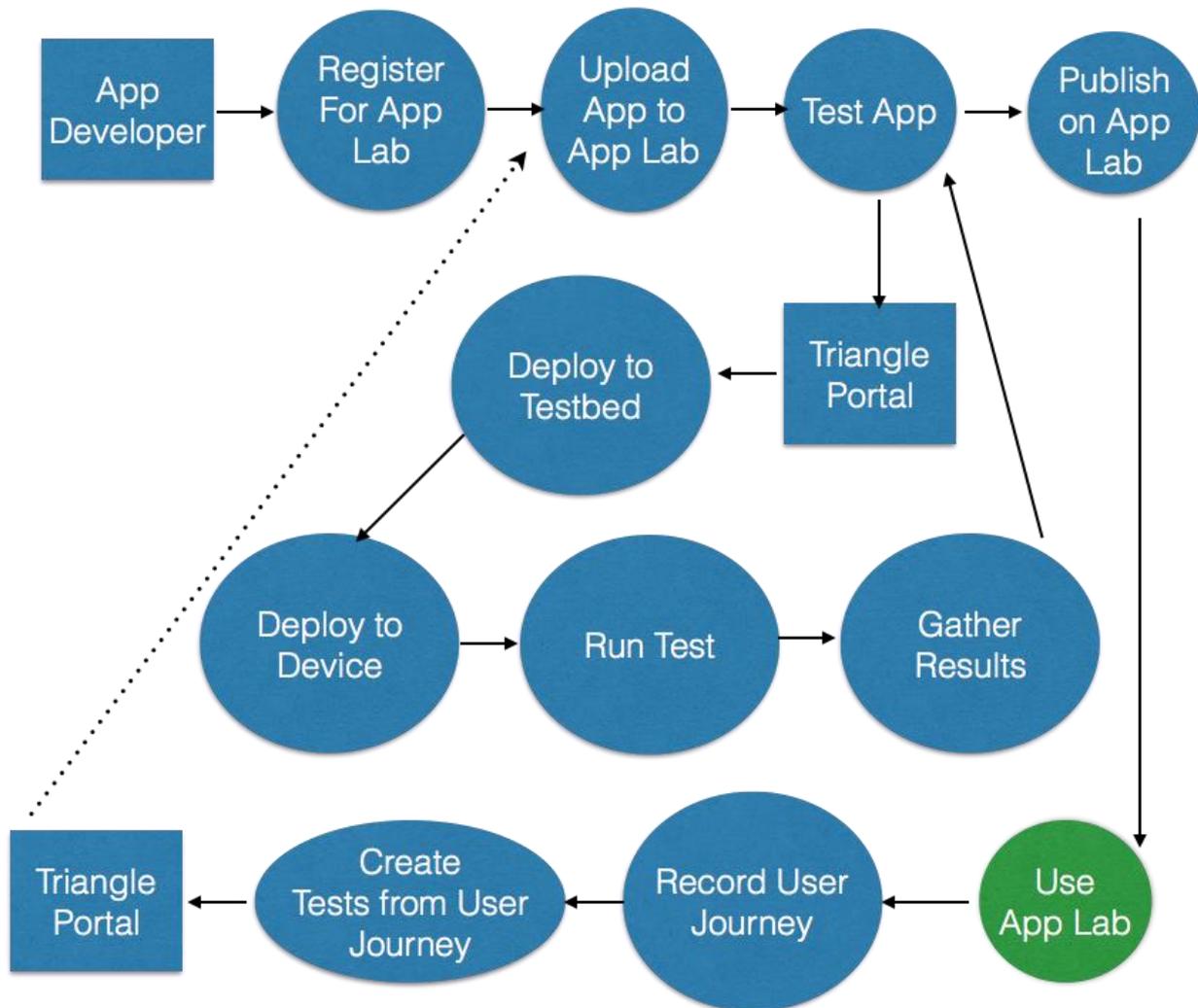


Figure 3 - App Lab Triangle Flow

QuaMotion are exploring the possibility of using the meta data generated from tools such as UXCam that describe user journeys and do use this meta data to generate automated test paths to execute on devices in the UMA Testbed. If such a technical integration is feasible, then it will be a great benefit towards more accurate testing based on real world usage of the applications.

3.8 TRIANGLE Portal integration with App Lab

Apps uploaded to App Lab can be automatically submitted to the UMA Testbed for testing. The plan is for this to be automated using, for example, FTP over a VPN connection, or using a web service. For the initial phase of the Triangle Testbed development during 2017 this is likely to be a manual process that is carried out by the tester assigned to each app that has been submitted to App Lab.



Document: ICT-688712-TRIANGLE/D5.2

Date: 01/09/2016

Dissemination: PU

Status: Final

Version: 1.0

TRIANGLE Submission Flow Steps:

1. app developer provides app
2. creation of application user flow (**using App Lab users for journey creation**)
3. select from a set of reference handsets
4. choose from a set of network scenarios
5. submit for testing
6. auto configure test environment
7. lab testing
8. evaluating the app against relevant KPIs
9. crowd / HCI testing (optional at UCL)
10. evaluating the app against relevant KPIs
11. reporting and Certification
- 12. Optional Deployment onto App Lab possible at this point for Human Feedback**



4 App Lab TRIANGLE Testbed Integration Summary

An overview of App Lab and the benefits of the platform were outlined.

The App Lab TRIANGLE Testbed integration proposal has been presented as the process to use both Triangle and UCL App Lab testbeds with the same app, but limited physical integration is expected to facilitate secure app transfer between App Lab and the Triangle Portal as outlined in Section 3.7.1.

Some aspects of App Lab for iOS distribution may be used by Testbed as part of app installation.

Apps on App Lab may use a controllable Wi-Fi and LTE network at UCL that would justify a VPS engine integration to allow more accurate device testing under different network conditions and to examine user's reaction to the UX and provide feedback to App Developers.

No deployment in Malaga is expected at this time as a physical installation does not make sense for a number of reasons illustrated in Section 3.6.

App Lab will use the services of the Testbed for some parts of the App Testing flow.

The key point is that user use of applications captured by the UXCam tool used by App Lab will be used to track user behavior that can be used to potentially create accurate user journeys that can be automated. We are currently investigating the feasibility of automatic data extraction from the UXCam feeds and translation into test scripts to be executed on devices in the UMA testbed. This may prove to be as good as, if not better than, any existing test cases that may have been supplied by the app developers.



Document: ICT-688712-TRIANGLE/D5.2

Date: 01/09/2016

Dissemination: PU

Status: Final

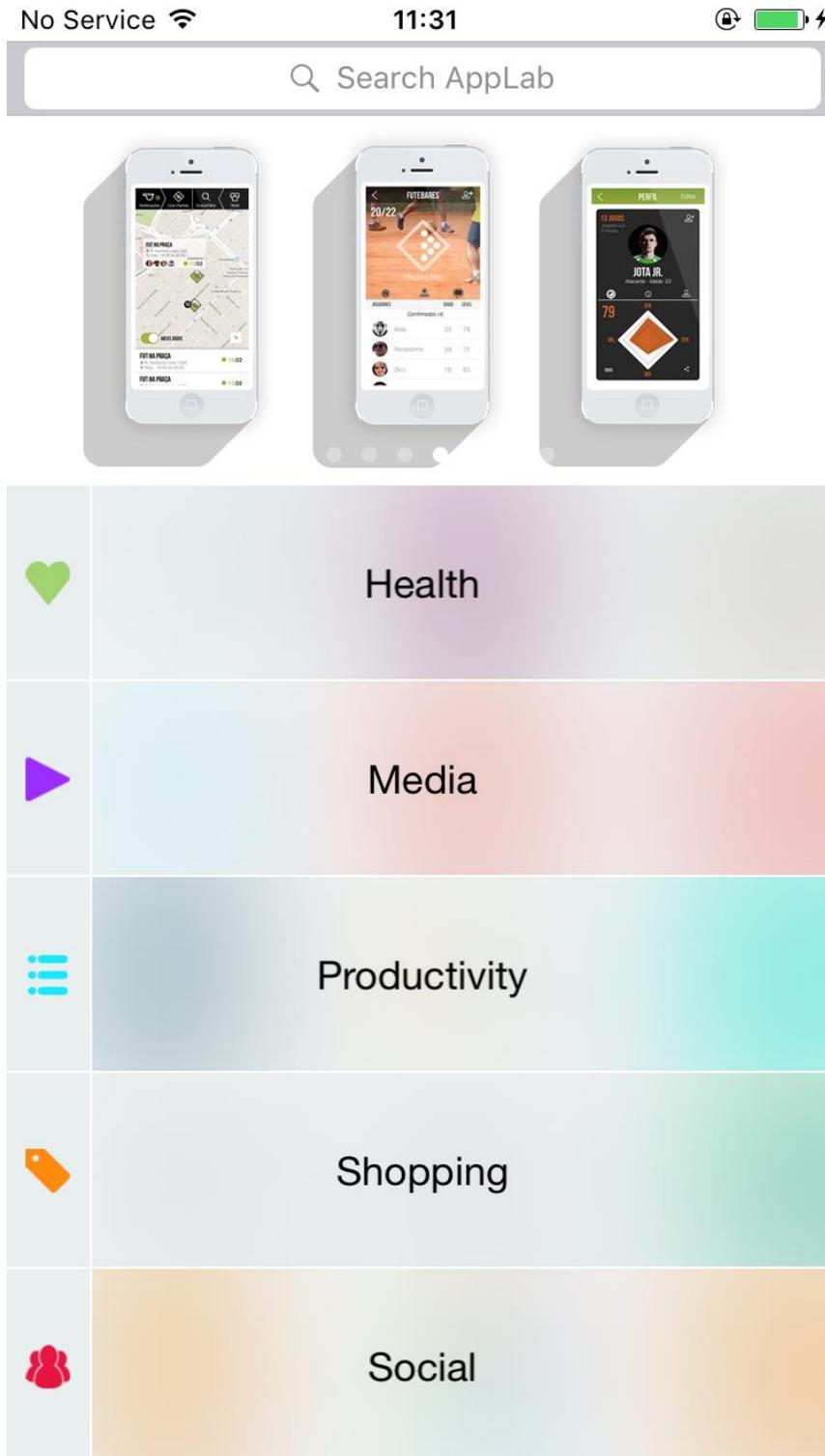
Version: 1.0

5 References

- [1] App Lab website <https://applab.ucl.ac.uk/>
- [2] UCL Innovation and Enterprise (formerly UCL Advances) <http://www.ucl.ac.uk/enterprise>
- [3] Apple iOS App Store <http://www.apple.com/itunes/>
- [4] Google Android Play Store <https://play.google.com/store/apps?hl=en>
- [5] Apple Developer Enterprise Program <https://developer.apple.com/programs/enterprise/>
- [6] UXCam User Experience improvement tool <http://uxcam.com/>
- [7] A/B Testing Wikipedia Entry https://en.wikipedia.org/wiki/A/B_testing
- [8] QuaMotion mobile app test automation website <http://mwc.quamotion.mobi/>
- [9] TestFairy iOS and Android Beta Testing solution <https://testfairy.com/>
- [10] Innovate UK <https://www.gov.uk/government/organisations/innovate-uk>
- [11] LabWiki <https://github.com/mytestbed/labwiki>



6 Annex 1. App Lab iOS Screenshots





No Service

11:34



Search AppLab



	 Kela Mental Resilience Health	 Viva la Vita - Fitness on... Health	 Footba ker Health
	Media		
	Productivity		
	 IStyleMyself Shopping	 Guystyler Shopping	
	Social		



No Service 11:32

IceCream.

George Berkowski
Productivity

Download

Description	Images	Review
-------------	--------	--------

IceCream is a clever camera app. It helps you to free up space on your iPhone for 250, 1000 or more photos in just a few taps. It is also a super fast, and simple, way to send 1, or 20 photos, in a safe way. And on top of that, all your photos are automatically organised by person, time and place. That's sweet.

FREE UP SPACE
IceCream allows you to FREE UP precious space on your iPhone. The app tells you how many photos you have space to take, and then allows you to FREE UP more space or storage without having to delete precious photos, videos or apps.

We do this in a clever powerful way that allows you to keep every single photo on your iPhone, and make it viewable, even when you are offline. IceCream allows you to generate beautiful, optimised versions of all your photos that stay on your iPhone, inside the IceCream app, while the much larger original photos, are stored in the IceCream Cloud. That means you can still find, view and share all your photos from your iPhone, but all the big photo files are safe in the IceCream Cloud where only you can access them.

SEND TONNES OF PHOTOS, FAST
The IceCream app is designed to send a lot of photos, fast. We know that you want to be able send and share photos faster, better and more simply ~ and have them conveniently organised automatically. Our unique IceCream technology allows you to send more photos, faster - and at the same time photos take up less precious storage space on your iPhone.

Lowdown for Meetings

IceCream.

Digimeal

Shopping



7 Annex 2. App Lab Portal Screenshots

The screenshot shows the App Lab developer portal interface. At the top, there is a navigation menu with options: Users, Apps, Add App, Approve for testing, Approve for publication, Create Test, and Tests. Below the navigation, the user is identified as 'UCL Decide' with options for 'Developer Docs' and 'Log out'. The main content area is divided into two sections: 'Not ready for approval' and 'Waiting for approval'. Each section contains a list of apps with their icons, names, categories, ratings, and links to Description, Binaries, and Media. The 'Waiting for approval' section also shows the number of active installs and the publication status for each app.

App Name	Category	Rating	Comments	Description	Binaries	Media
Test App N...	Entertainment	☆☆☆☆☆	(0) Comments	Description	0 Active Installs	Media
Viva la VL...	Health	☆☆☆☆☆	(0) Comments	Description	0 Active Installs	Media
Waiting for approval						
5-TILES ke...	Utility	☆☆☆☆☆	(0) Comments	Description	0 Active Installs	Media
Antix	Media	☆☆☆☆☆	(0) Comments	Description	0 Active Installs	Media
Bubbal	Social	☆☆☆☆☆	(0) Comments	Description	17 Active Installs Published	Media
Citrus Sat...	UCL Apps	☆☆☆☆☆	(0) Comments	Description	0 Active Installs	Media
flashbackr	Productivity	☆☆☆☆☆	(0) Comments	Description	0 Active Installs	Media



App Lab
Users Apps Add App Approve for testing Approve for publication Create Test Tests

User: UCL Decide Developer Docs Log out

App	Test	Finished tests	Trials	Reject / Approve
UCL Audio... UCL Apps	2 Total test	27/09/2014 0 test(s) failed	No Trials	<input type="button" value="Reject"/> <input type="button" value="Approve for publication"/>
UCL Golf UCL Apps	2 Total test	27/09/2014 0 test(s) failed	No Trials	<input type="button" value="Reject"/> <input type="button" value="Approve for publication"/>

© UCL 2014



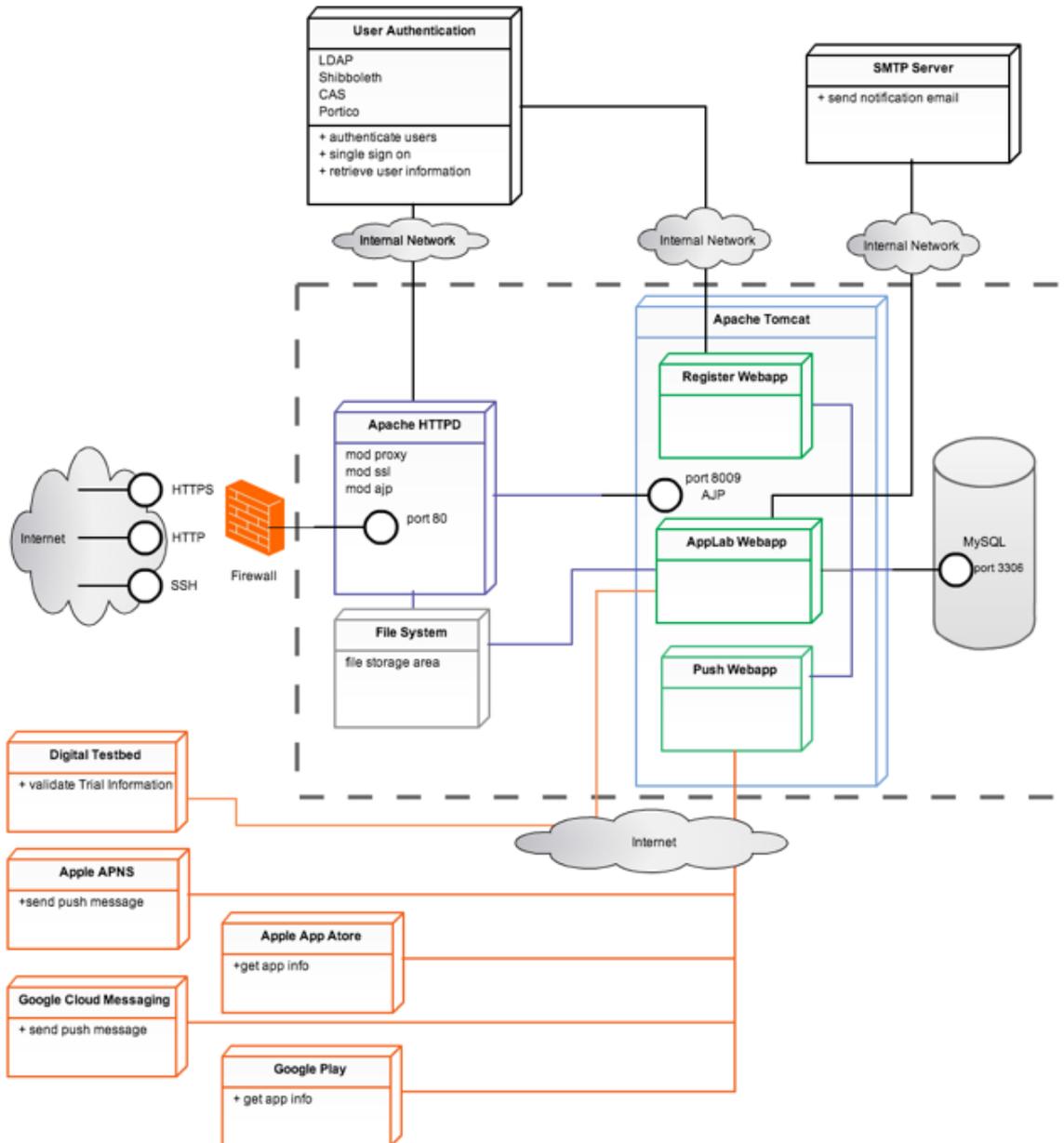
The screenshot shows the 'Approve for testing' interface in App Lab. It features a navigation bar with 'Users', 'Apps', 'Add App', 'Approve for testing', 'Approve for publication', 'Create Test', and 'Tests'. Below the navigation bar, there are links for 'User: UCL Decide', 'Developer Docs', and 'Log out'. The main content area is titled 'Approve for testing' and is divided into three columns:

- Select tests groups:** Drag and drop over the version to add test case group to version. It lists four groups: Android Technical Test, Android UX & UI Test, IOS Technical Test, and IOS UX & UI Test.
- Versions:** Click on the arrow to see the assignment. It shows two versions: 'flashback 0.0.1 (iPhone)' and 'rendeevoo 2.0.1 (iPhone)'. The 'rendeevoo' version is currently selected and assigned to Niall Roche. There are 'Reject' and 'Start test' buttons for the 'rendeevoo' version.
- Assign testers:** Drag and drop over the test case group dropped. It lists several testers: Ann Chan, Daniel Fozzati, John doe, Jules Spa, Niall, Niall Roche, and Nicolas Arrellaga.

© UCL 2014



8 Annex 3. App Lab Software Architecture





Document: ICT-688712-TRIANGLE/D5.2

Date: 01/09/2016

Dissemination: PU

Status: Final

Version: 1.0

9 Annex 4. VELOX VPS Engine

The VELOX VPS Engine API provides developers with a set of resources that allow the listing and control of VPS services. The API is a set of JSON encoded requests and responses sent over a TCP connection. All service availability is bound to the API Key used.

In order to use the VELOX API an application must:

1. Create a TCP connection to known IP address/port (provided by local operator)
2. Write Request (as a single text line, new line ends a request)
3. Read Response (sent as a single line) Connections are terminated on the VELOX side after sending the response All Requests must use the API key generated by the local operator VELOX.

In this scenario the client application is considered to be any application that uses the VELOX API to access VELOX services.

Modify and Run requests are considered optional since not all circumstances will require their use.

While the List request is the first to be executed it is not mandatory before every Trigger request, the use of the List request should be done as deemed necessary in order to check for any service library changes.

