

# Jenkins - Jobs and Pipelines

## Build process

### Pipelines :

Found under the pipeline tab.

The pipelines are configured using the Multibranch pipeline option available in Jenkins 2.x. The code for the pipeline execution steps is a part of the App code itself. There are Jenkinsfiles in the home folder of the project and these Jenkinsfiles intern call groovy script files present within the pipeline folder.

<http://192.168.2.201:8086/view/Pipelines/>

**Jenkins**

Search:  LOG IN

Jenkins > Pipelines

ENABLE AUTO REFRESH

**Pipelines for GHS Xamarin Apps**

Note : LIVE and OPS build upon commits. Manually trigger Store5 and Store6.

Admin-Tools All Build Status CI - DEV - Build Status CI-DEV CI-MAS CI-REL Develop Branch

Digital Clubcard - Build Status Master Branch Nightly-Builds OnDemand-Builds **Pipelines** Regression Jobs Release Branch

Sanity Jobs

S	W	Name ↓	Last Success	Last Failure	Last Duration	Progress
		GHS Android - Xamarin	12 min - <a href="#">log</a>	N/A	5.4 sec	
		GHS Android - Xamarin - OPS	1 min 35 sec - <a href="#">log</a>	N/A	8.1 sec	
		GHS Android - Xamarin - Store5	2 mo 7 days - <a href="#">log</a>	N/A	7.6 sec	
		GHS Android - Xamarin - Store6	2 mo 8 days - <a href="#">log</a>	N/A	3 min 49 sec	
		GHS-iOS - Xamarin	N/A	1 min 35 sec - <a href="#">log</a>	9.3 sec	
		GHS-iOS - Xamarin - OPS	16 min - <a href="#">log</a>	N/A	6.3 sec	
		GHS-iOS - Xamarin - STORE5	2 mo 7 days - <a href="#">log</a>	N/A	12 sec	
		GHS-iOS - Xamarin - STORE6	2 mo 7 days - <a href="#">log</a>	N/A	22 sec	

**Build Queue (1)**

DC-Android-Sanity > AndroidAppiumCucumber

**Build Executor Status**

**master**

- 1 Idle
- 2 Idle
- 3 Idle
- 4 Idle
- 5 Idle

**Android\_Phone\_VM** (offline)

**Android\_Tablet\_VM** (offline)

The following are the pipeline jobs setup in Jenkins:

### GHS Android

➤ Release -

<http://192.168.2.201:8086/view/Pipelines/job/GHS%20Android%20->

## %20Xamarin/

S	W	Name	Last Success	Last Failure	Last Duration	Progress
✓	⚙️	develop	2 days 14 hr - #898	N/A	15 min	▶
✓	⚙️	master	4 days 0 hr - #3	N/A	1 hr 12 min	▶
✓	⚙️	release/TescoRelease	4 days 0 hr - #21	10 days - #8	1 hr 14 min	▶

- **OPS** - <http://192.168.2.201:8086/view/Pipelines/job/GHS%20Android%20-%20Xamarin%20-%20OPS/>
- **Store 5** - <http://192.168.2.201:8086/view/Pipelines/job/GHS%20Android%20-%20Xamarin%20-%20Store5/>
- **Store 6** - <http://192.168.2.201:8086/view/Pipelines/job/GHS%20Android%20-%20Xamarin%20-%20Store6/>

## GHS iOS

- **Release** - <http://192.168.2.201:8086/view/Pipelines/job/GHS-iOS%20-%20Xamarin/>
- **OPS** - <http://192.168.2.201:8086/view/Pipelines/job/GHS-iOS%20-%20Xamarin%20-%20OPS/>
- **Store 5** - <http://192.168.2.201:8086/view/Pipelines/job/GHS-iOS%20-%20Xamarin%20-%20STORE5/>
- **Store 6** - <http://192.168.2.201:8086/view/Pipelines/job/GHS-iOS%20-%20Xamarin%20-%20STORE6/>

In all the above multi branch pipelines, the jobs are listed by branch names. The builds would be available within each branch(job). The pipelines are configured to run only for the develop, release and master branches. These pipelines will poll the GitHub repos every 20mins to check if there are any changes. If there are any

changes, builds for that particular branch (from dev, release or master) will be triggered automatically. Builds will be uploaded to hockey app as a part of the pipeline.

## On-Demand Build Jobs :

These jobs handle on-demand builds. They include feature branch builds and builds for other configurations like OPS Store5 and Store6 if the pipeline jobs are failing.

<http://192.168.2.201:8086>

On Demand Build View - (dev-od is DEVELOP branch) - (rel-od is RELEASE branch) - (mas-od is MASTER branch) - Manual trigger is allowed.

Admin-Tools | All | Build Status | CI - DEV - Build Status | CI-DEV | CI-MAS | CI-REL | Develop Branch

Digital Clubcard - Build Status | Master Branch | Nightly-Builds | **OnDemand-Builds** | Pipelines | Regression Jobs | Release Branch

Sanity Jobs +

S	W	Name ↓	Last Success	Last Failure	Last Duration	Progress
✓		Custom-DC-Native-Build	3 days 11 hr - #324	N/A	9 min 26 sec	▶
✓		Custom-GHS-Build	1 mo 1 day - #409	1 mo 2 days - #408	19 min	▶
✓		Custom-GHS-Build-Debug	3 mo 26 days - #77	3 mo 26 days - #76	7 min 39 sec	▶
✓		Custom-GHS-Build-OPS	1 mo 0 days - #70	1 mo 0 days - #69	13 min	▶
✓		Custom-StoreEmpower-Build	1 yr 1 mo - #6	1 yr 1 mo - #5	1 min 8 sec	▶
✓		Custom-StoreEmpowerment-Build	10 days - #8	10 days - #7	2 min 59 sec	▶
✓		Custom-Upload-to-Hockey	2 yr 1 mo - #7	N/A	32 sec	▶
✓		Custom_JGHS_Build	10 mo - #61	10 mo - #60	3 min 13 sec	▶

1. **Custom-GHS-Build:** This job provides for generating builds from a feature branch. “ReleaseWIP” configuration for Android and “Release” configuration for iOS.
2. **Custom-GHS-Build-Debug:** This job provides for generation of feature branch builds. “Debug” configuration for both Android and iOS.
3. **Custom-GHS-Build-OPS:** This job provides for generating builds from a feature branch. “OPS” config for Android and “OPSRelease” config for iOS.
4. **GHS-Android-Build-dev-od:** This job would be the backup job for generating OPS, Store5 and Store6 builds for Android from develop branch if the pipeline jobs mentioned above are not working fine.

5. **GHS-Android-Build-rel-od**: Same as above. But for “release” branch.
6. **GHS-Android-Build-mas-od**: Same as above. For “master” branch.
7. **GHS-iOS-Build-dev-od**: This job would be the backup job for generating OPS, Store5 and Store6 builds of iOS from develop branch if the pipeline jobs mentioned above are not working fine.
8. **GHS-iOS-Build-rel-od**: Same as above. But for “release” branch.
9. **GHS-iOS-Build-mas-od**: Same as above. But for “master” branch.

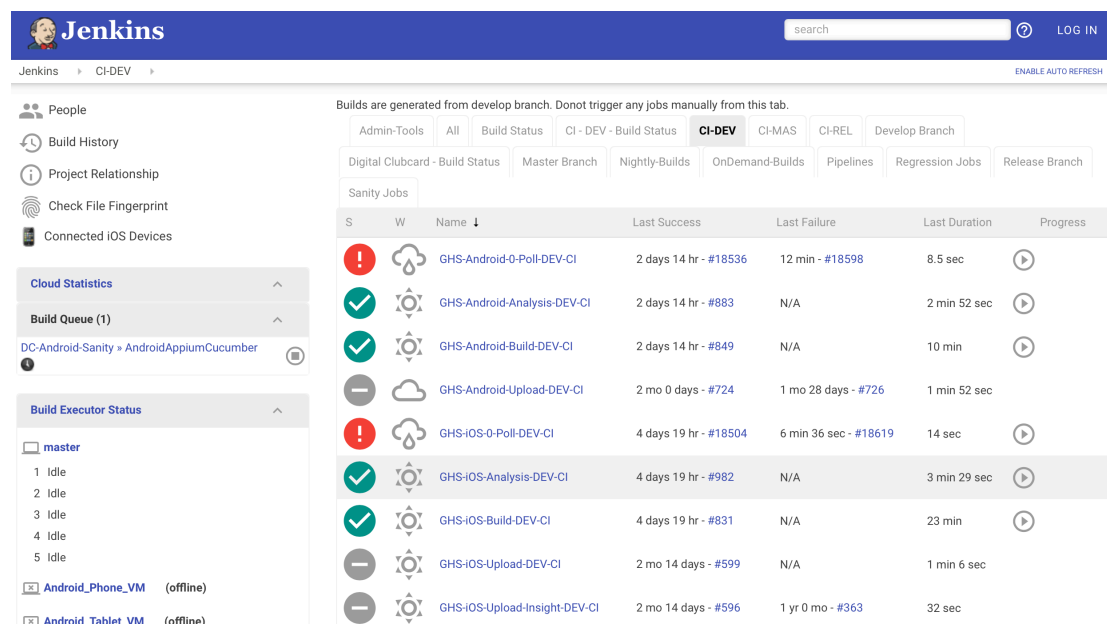
## Other Build Jobs:

Apart from the pipelines, there are build jobs which can be used as a backup if the pipelines are not working as expected.

These are the old style pipelines where individual jobs handle a part of the pipeline and these jobs are tied together as a pipeline. They are configured for “develop”, “release/TescoRelease” and “master” branches respectively.

## Develop Branch (Found under “CI-DEV” tab)

<http://192.168.2.201:8086/view/CI-DEV/>



The screenshot shows the Jenkins web interface for the CI-DEV tab. The left sidebar contains navigation links: People, Build History, Project Relationship, Check File Fingerprint, Connected iOS Devices, Cloud Statistics, Build Queue (1), and Build Executor Status. The main content area displays a table of build jobs. The table has columns for Status (S), Workflow (W), Name, Last Success, Last Failure, Last Duration, and Progress. The jobs are listed in descending order of last success time.

S	W	Name ↓	Last Success	Last Failure	Last Duration	Progress
!	☁	GHS-Android-0-Poll-DEV-CI	2 days 14 hr - #18536	12 min - #18598	8.5 sec	▶
✓	⚙	GHS-Android-Analysis-DEV-CI	2 days 14 hr - #883	N/A	2 min 52 sec	▶
✓	⚙	GHS-Android-Build-DEV-CI	2 days 14 hr - #849	N/A	10 min	▶
—	☁	GHS-Android-Upload-DEV-CI	2 mo 0 days - #724	1 mo 28 days - #726	1 min 52 sec	
!	☁	GHS-iOS-0-Poll-DEV-CI	4 days 19 hr - #18504	6 min 36 sec - #18619	14 sec	▶
✓	⚙	GHS-iOS-Analysis-DEV-CI	4 days 19 hr - #982	N/A	3 min 29 sec	▶
✓	⚙	GHS-iOS-Build-DEV-CI	4 days 19 hr - #831	N/A	23 min	▶
—	⚙	GHS-iOS-Upload-DEV-CI	2 mo 14 days - #599	N/A	1 min 6 sec	
—	⚙	GHS-iOS-Upload-Insight-DEV-CI	2 mo 14 days - #596	1 yr 0 mo - #363	32 sec	

## - Android

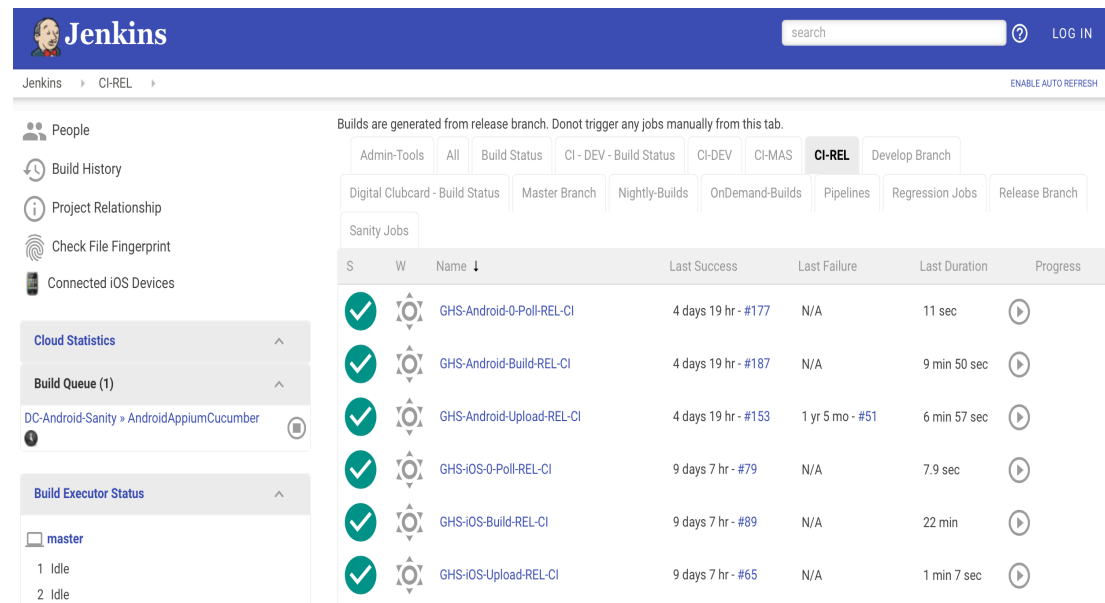
1. **GHS-Android-0-Poll-DEV-CI:** Runs every hour. Checks for any new commits. If yes, triggers GHS-Android-Analysis-DEV-CI.
2. **GHS-Android-Analysis-DEV-CI:** Triggered automatically by the above poll job. This job handles the code analysis part which is run using SonarQube.
3. **GHS-Android-Build-DEV-CI:** Triggered automatically by the above poll job. Builds GHS-Android application with “Release” configuration.
4. **GHS-Android-Upload-DEV-CI:** Triggered if the above build job succeeds. Uploads the latest Release build on Dev branch. (Currently disabled as upload is being handled by the pipeline job)

## - iOS

1. **GHS-iOS-0-Poll-DEV-CI:** Runs every hour. Checks for any new commits. If yes, triggers GHS-iOS-Analysis-DEV-CI.
2. **GHS-iOS-Analysis-DEV-CI:** Triggered automatically by the above poll job. This job handles the code analysis part which is run using SonarQube.
3. **GHS-iOS-Build-DEV-CI:** Triggered automatically by the above poll job. Builds GHS-Android application with “LiveRelease” configuration.
4. **GHS-iOS-Upload-DEV-CI:** Triggered if the above build job succeeds. Uploads the latest LiveRelease build on Dev branch. (Currently disabled as upload is being handled by the pipeline job)

## Release Branch (Found under “CI-REL” tab)

<http://192.168.2.201:8086/view/CI-REL/>



The screenshot shows the Jenkins web interface for the 'CI-REL' tab. The left sidebar contains navigation links: People, Build History, Project Relationship, Check File Fingerprint, Connected iOS Devices, Cloud Statistics, Build Queue (1), DC-Android-Sanity > AndroidAppiumCucumber, Build Executor Status, and master. The main content area displays a table of build jobs. The table has columns for S (Success), W (Warning), Name, Last Success, Last Failure, Last Duration, and Progress. The jobs listed are:

S	W	Name	Last Success	Last Failure	Last Duration	Progress
✓		GHS-Android-0-Poll-REL-CI	4 days 19 hr - #177	N/A	11 sec	▶
✓		GHS-Android-Build-REL-CI	4 days 19 hr - #187	N/A	9 min 50 sec	▶
✓		GHS-Android-Upload-REL-CI	4 days 19 hr - #153	1 yr 5 mo - #51	6 min 57 sec	▶
✓		GHS-iOS-0-Poll-REL-CI	9 days 7 hr - #79	N/A	7.9 sec	▶
✓		GHS-iOS-Build-REL-CI	9 days 7 hr - #89	N/A	22 min	▶
✓		GHS-iOS-Upload-REL-CI	9 days 7 hr - #65	N/A	1 min 7 sec	▶

### - Android

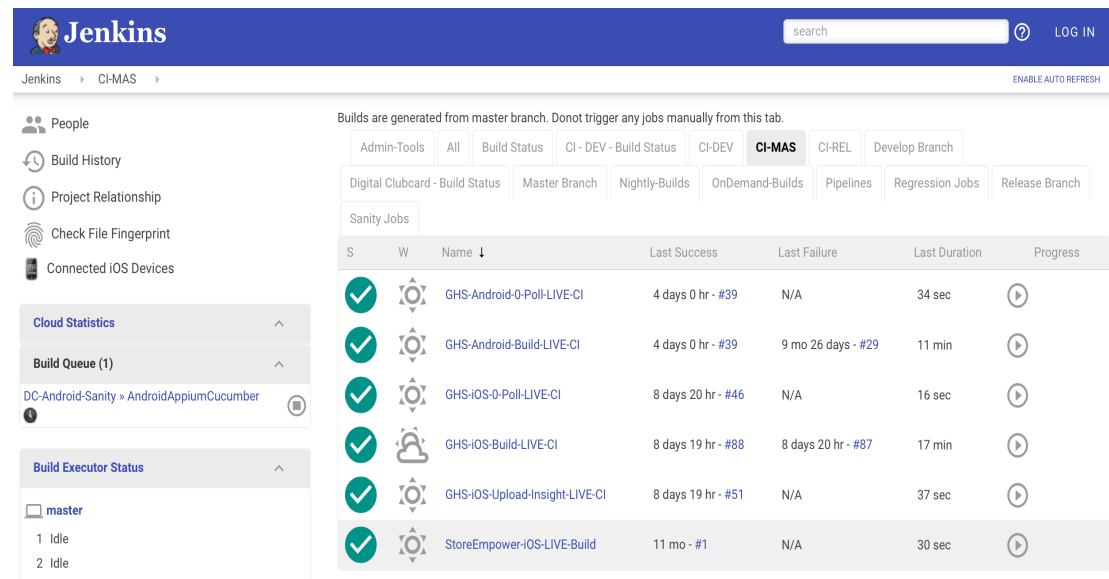
1. **GHS-Android-0-Poll-REL-CI**: Runs once in a week. Checks for any new commits. If yes, triggers GHS-Android-Build-REL-CI.
2. **GHS-Android-Build-REL-CI**: Triggered automatically by the above poll job. Builds GHS-Android application with “Release” configuration on the release branch.
3. **GHS-Android-Upload-REL-CI**: Triggered if the above build job succeeds. Uploads the latest Release build on Release branch to hockeyapp.

### - iOS

1. **GHS-iOS-0-Poll-REL-CI**: Runs once in a week. Checks for any new commits. If yes, triggers GHS-iOS-Build-REL-CI.
2. **GHS-iOS-Build-REL-CI**: Triggered automatically by the above poll job. Builds GHS-iOS application with “LiveRelease” configuration on the release branch.
3. **GHS-iOS-Upload-REL-CI**: Triggered if the above build job succeeds. Uploads the latest LiveRelease build on Release branch to hockeyapp.

## Master Branch (Found under “CI-MAS” tab)

<http://192.168.2.201:8086/view/CI-MAS/>



The screenshot shows the Jenkins web interface. The top navigation bar includes the Jenkins logo, a search bar, and a 'LOG IN' button. Below the navigation bar, the breadcrumb trail shows 'Jenkins > CI-MAS >'. The left sidebar contains various links: People, Build History, Project Relationship, Check File Fingerprint, Connected iOS Devices, Cloud Statistics, Build Queue (1), DC-Android-Sanity > AndroidAppiumCucumber, and Build Executor Status. The main content area displays a message: 'Builds are generated from master branch. Donot trigger any jobs manually from this tab.' Below this message is a tabbed interface with tabs for Admin-Tools, All, Build Status, CI - DEV - Build Status, CI-DEV, CI-MAS (selected), CI-REL, and Develop Branch. Under the CI-MAS tab, there are sub-tabs for Digital Clubcard - Build Status, Master Branch, Nightly-Builds, OnDemand-Builds, Pipelines, Regression Jobs, and Release Branch. The Master Branch sub-tab is selected, showing a table of builds. The table has columns for S (Status), W (Webhook), Name, Last Success, Last Failure, Last Duration, and Progress. The builds listed are:

S	W	Name	Last Success	Last Failure	Last Duration	Progress
✓	⚙️	GHS-Android-0-Poll-LIVE-CI	4 days 0 hr - #39	N/A	34 sec	▶
✓	⚙️	GHS-Android-Build-LIVE-CI	4 days 0 hr - #39	9 mo 26 days - #29	11 min	▶
✓	⚙️	GHS-iOS-0-Poll-LIVE-CI	8 days 20 hr - #46	N/A	16 sec	▶
✓	⚙️	GHS-iOS-Build-LIVE-CI	8 days 19 hr - #88	8 days 20 hr - #87	17 min	▶
✓	⚙️	GHS-iOS-Upload-Insight-LIVE-CI	8 days 19 hr - #51	N/A	37 sec	▶
✓	⚙️	StoreEmpower-iOS-LIVE-Build	11 mo - #1	N/A	30 sec	▶

### - Android

1. **GHS-Android-0-Poll-LIVE-CI**: Runs once in a week. Checks for any new commits. If yes, triggers GHS-Android-Build-LIVE-CI.
2. **GHS-Android-Build-LIVE-CI**: Triggered automatically by the above poll job. Builds GHS-Android application with “Release” configuration.

### - iOS

1. **GHS-iOS-0-Poll-LIVE-CI**: Runs once in a week. Checks for any new commits. If yes, triggers GHS-iOS-Build-LIVE-CI.
2. **GHS-iOS-Build-LIVE-CI**: Triggered automatically by the above poll job. Builds GHS-Android application with “Appstore” configuration.

Jobs not automated for master branch as commits are done only during a release.

# Test process

Test automation is setup in Jenkins. This includes the creation of build, carrying out the tests and then mailing the results to the team as well as displaying the same on the Jenkins job page. The jobs related to tests are grouped into folders under the “Sanity” tab in Jenkins.

## GHS Android

The test automation is available in 3 forms:

- Sanity
- Priority
- Regression

<http://192.168.2.201:8086/view/Sanity%20Jobs/job/GHS-Android/>

The screenshot shows the Jenkins web interface for the 'GHS-Android' job. The top header is blue with the Jenkins logo and a search bar. Below the header, there's a breadcrumb trail: Jenkins > Sanity Jobs > GHS-Android. The main content area is titled 'GHS-Android' and shows a table of builds. The left sidebar contains navigation links: Up, Status, People, Build History, Project Relationship, and Check File Fingerprint. Below these are sections for 'Cloud Statistics', 'Build Queue (1)', and 'Build Executor Status'.

S	W	Name ↓	Last Success	Last Failure	Last Duration	Progress
✓	⚙️	GHS-Android-Sanity-Mailer	37 min - #1179	N/A	9.2 sec	▶
—	⚙️	GHS_Android_Priority_Build	1 mo 0 days - #168	N/A	17 min	
—	⚙️	GHS_Android_Priority_Phone	1 yr 1 mo - #58	N/A	47 min	
—	☁️	GHS_Android_Priority_Tablet	1 mo 21 days - #470	1 mo 18 days - #474	39 min	
—	⚙️	GHS_Android_Regression_Build_n	1 mo 1 day - #351	N/A	10 min	
!	☁️	GHS_Android_Regression_Phone_n	10 mo - #114	10 mo - #117	3 hr 25 min	▶
!	☁️	GHS_Android_Regression_Tablet_n	10 mo - #117	10 mo - #127	6 hr 30 min	▶
✓	⚙️	GHS_Android_Sanity_Build_n	1 hr 41 min - #1856	N/A	5 min 8 sec	▶
⚠️	⚙️	GHS_Android_Sanity_Phone	6 mo 0 days - #8	N/A	1 hr 29 min	▶
✓	⚙️	GHS_Android_Sanity_Tablet	1 hr 36 min - #1364	N/A	58 min	▶



## Sanity –

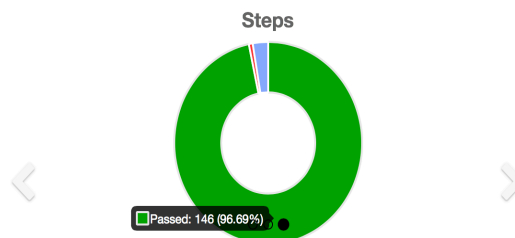
1. **GHS\_Android\_Sanity\_Build\_n**: This job builds the application afresh for every run. Sanity is currently run on Master builds.
2. **GHS\_Android\_Sanity\_Tablet**: This job would run the sanity test automation on a tablet. It is integrated with the Jacoco and Cobertura plugin which would help display the results in a graphical form

Cucumber Report	Jenkins	Previous results	Latest results	Features	Tags	Steps	Trends
-----------------	---------	------------------	----------------	----------	------	-------	--------

Project	Number	Date
GHS_Android_Sanity_Tablet	1364	21 Aug 2017, 12:03

### Features Statistics

The following graphs show passing and failing statistics for features



Feature	Steps						Scenarios			Features	
	Passed	Failed	Skipped	Pending	Undefined	Total	Passed	Failed	Total	Duration	Status
Sanity	146	1	4	0	0	151	17	1	18	42m 34s 986ms	Failed
1	146	1	4	0	0	151	17	1	18	42m 34s 986ms	
	96.69%	0.66%	2.65%	0.00%	0.00%		94.44%	5.56%			0.00%

### Cucumber Report

3. **GHS\_Android\_Sanity\_Phone**: Same as above. But for Phone.
4. **GHS-Android-Sanity-Mailer**: This job runs a powershell script that helps sending the test results in HTML format to the whole team.

### GHS Mobile App Sanity - 21-08-2017

Tablet					
	Scenarios	Passed	Failed	Pending	Pass Percentage
Sanity	18	17	1	0	94
Rerun-1	1	1	0	0	100
Totals	18	18	0	0	100

----- 100 % tests passed -----

For detailed Sanity reports : [Click here](#)

## **Priority –**

1. **GHS\_Android\_Priority\_Build\_n:** This job builds the application afresh for every run. Priority is currently run on Develop builds.
2. **GHS\_Android\_Priority\_Tablet:** This job would run the priority test automation on a tablet. It is integrated with the Jacoco and Cobertura plugin which would help display the results in a graphical format.
3. **GHS\_Android\_Priority\_Phone:** Same as above. But for Phone.

## **Regression –**

1. **GHS\_Android\_Regression\_Build\_n:** This job builds the application afresh for every run. Priority is currently run on Develop builds.
2. **GHS\_Android\_Regression\_Tablet:** This job would run the priority test automation on a tablet. It is integrated with the Jacoco and Cobertura plugin which would help display the results in a graphical format.
3. **GHS\_Android\_Regression\_Phone:** Same as above. But for Phone.