

BuildPan x Linkites(IT Company)

USE CASE

Integration CI/CD Cloud

Aug 06, 2020

SynopsisIR

This client is well known for his consistency in making seamless IT solutions. They were working on various projects on different platforms

However, the same client was keen on adopting new technologies to keep pace with the modern times. Specifically, during COVID when its team was working remotely they needed a solution to keep pace in their project without any hick-ups.

BuildPan helped the client sail smoothly through the transformation with its customised solution.

Challenges/Problems

Problems Faced by Mobile Application Developers ::

Buildpan solves these 5 major problems faced by developers during the process of mobile application development.

1. [Manual Installation of Build](#) - On an average 15 mins are spent on manually downloading the build to device.
2. [Deployment on iTunes and Google Play Store!](#) - Deployment of apps on app stores and play stores with quality code.

3. **Sending builds for QA and getting bug reports!** - Sending .ipa & .apk files to QA has always been a pain point for developers and getting bug reports in the form of excel has created a transparency issue.
4. **Finding UDID to send the build** - Developing iOS apps require the device ID(alphanumeric combination of 24+ alphabets and numbers) to share build(.ipa file) which becomes difficult for a developer to ask every time to the client / QA / user.
5. **Cloud Platform** - Teams working remotely find it difficult to work together.

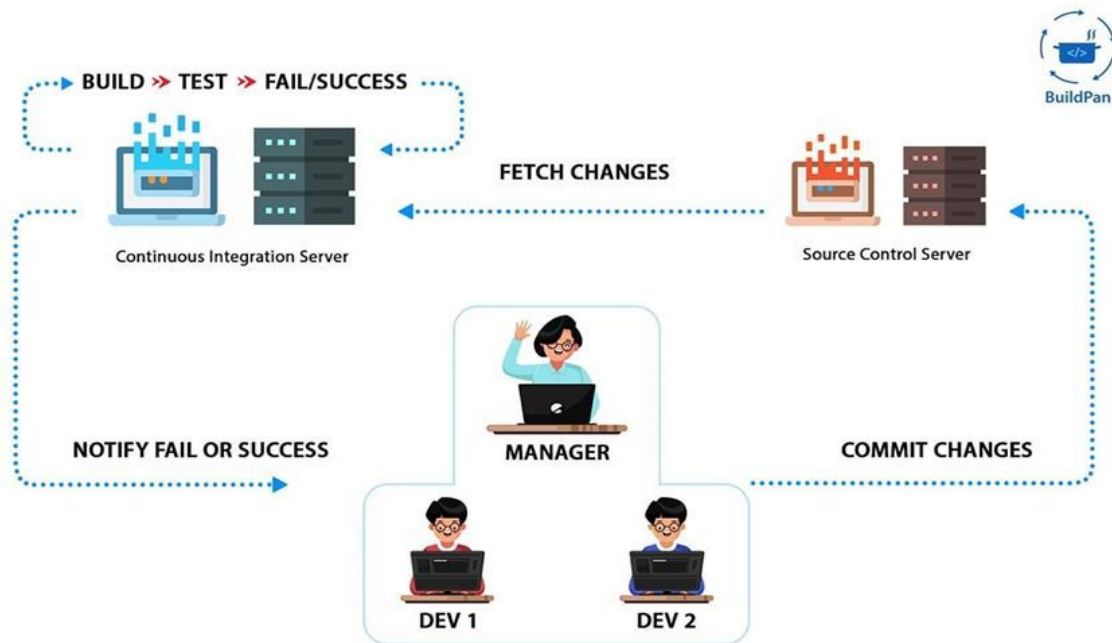
Product Overview

BuildPan- An AI cloud based CI/CD mobile application development platform automating the manual process to build , test & deploy mobile applications.

BuildPan CI/CD is an AI Cloud based engine shifting the manual intervention to automated execution in app building. It allows the team to rapidly build, test and deploy quality apps in the market. Our mission is to give developers the power to build and deploy tested apps in production environments.

Buildpan aims at shaping the mobile application development industry by making the developers meet the client requirements with accuracy and velocity focussing more on the productive side rather than being stuck at manual intervention process.

“Leveraging and Shaping the entire mobile application development industry”



Goals/Scope

Goals achieved by buildpan usage ::

1. Fewer handoffs:

The more handoffs in your development pipeline, the more points of failure and the more complexity you add to the process.

2. Increased development speed:

All stages of development are faster with CI/CD. Faster iterations throughout the process make every team more efficient and developers can move onto other projects confidently.

3. More deployments:

4

Releases that happened every couple of weeks or more, can now be pushed up to six or more times a day.

4. Faster testing:

One of the more time-consuming parts of the development workflow is removed and developers can work on other high-value projects. Automated testing lets teams get feedback sooner and fail early rather than catching bugs in production – or worse – in the final release.

5. Fewer bugs:

With automated testing throughout the development process, bugs are caught as they arise and rolled back without ever making it into master. This ensures better code quality overall and that every release works just as intended.

6. Improved compliance:

Compliance tasks can be incorporated into the development lifecycle, reducing the risk of releasing non-compliant applications. Automated compliance makes it easier to complete audits and can prevent costly mistakes (especially in highly-regulated industries).

7. More time to innovate:

Less time spent on integration maintenance and undifferentiated IT spend means resources can be directed elsewhere.

8. Happier developers:

Developers can work confidently and fix things quickly instead of waiting for weeks to learn there was a bug

9. Reduced overhead spending:

Organisations can only have so many developers at one time. Development hours are many times billable hours and manually deploying or testing code can bust IT budgets. An automated workflow reduces manual tasks and makes budgets more efficient.

10. Consistent processes:

Having more automation in the development workflow means nobody ever forgets a step in the process. Builds are more consistent, it's easier to train new developers, and organisations have more control over how things are built and when they get released.

Service

Integrated BuildPan CI/CD cloud solution for building and deploying mobile apps (Swift 4.2 above and Kotlin) dated 23rd Feb' @2020 till date.

Projects ::

1. OTT platform mobile application (iOS/Android)
2. E-commerce App (iOS/Android)
3. Media & Entertainment Mobile App
4. Social Media (iOS/Android)

And many more.....

Buildpan Solution

1. Frictionless Setup:

Set up a comprehensive continuous integration system for your iOS/Android apps in minutes.

1. AI Beta Testing:

Perform Beta check by running our machine learning algorithm on a click.

2. Effortless and Auto Deployments:

Deployments on a click to app stores.

3. Sample Code:

Our code lib will give you step by step development for your very first mobile/ web app

4. App Integration:

Share build through mail or slack by integrating your slack or email account.

5. Auto Bug Reports:

An auto bug report to the developer after QA testing.

6. Remote Access

Work remotely. Push your code to BuildPan Cloud, it will take care of the rest.

Timeline for POC and Production

POC	Timeline	Production	Timeline	Status
Setting Cloud server	5hours	Setting Cloud server	5hours	QA Check ok
Environment	2 hours	Environment	2-3 hours	QA Check ok
Creating accounts	15mins	Creating accounts	15mins	QA Check ok
Adding Repo	5mins	Adding Repo	5 mins	QA Check ok
Connecting with Linkites Itunes and Google Play	5mins	Connecting with Linkites Itunes and Google Play	5 mins	QA Check ok
Deployment	30mins	Deployment	30-40mins	QA Check ok
Total hours	8hrs	Total hours	9hrs	

Platform Usage:

- Projects - 7 repo Added
- Total mins - 680 mins total
- Team - 17 members added
- Per repo mins - 90mins on an average by single repo
- No of Builds per day - 8-10

Metrics:

Measurables	(A) BuildPan	(B) Manual
Adding repo	5mins	up/down time 1hour
Fetching repo	15-30mins	Downloading time
Setting up the project	5mins	1 hour average
Build time	5-15mins average	15-20mins average
Agile remote work	Yes	No
Sending build QA	On a click	Build and add device time consuming
Crash Report	On a click	Manual and 3rd party tool
Rectify and Fix time	On a click	Time consuming
Final QA check build	On a click	Manual and Time Consuming
Deployment staging/live	On a click	1-2hours
Production %	400% higher in comparison to B	LOW

Key Benefits:

- ADLC process became 70% fast
- Developers Multitasking increased from 1 or 2 projects to 3 -4 projects
- SLA driven business delivery meeting the client requirements on desired time lines for 9 out of 10 projects reaching 90% efficiency.
- 40% reduction in man hours
- 25% cost reduction

