

Seminar Report

M.A.U.I

Allen Bose

August 3, 2025

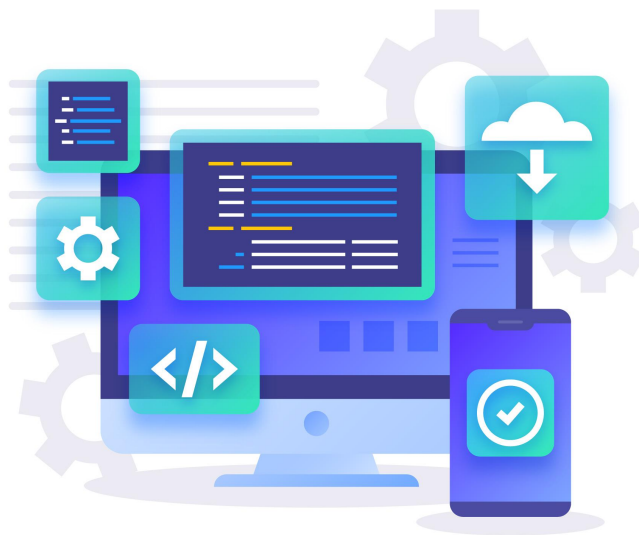
Contents

1	Cross Platform Applications	1
2	Cross Platform Frameworks	2
3	Cross-Platform frameworks of today	3
3.1	React Native	3
3.2	Flutter	4
4	Reference Links	5

1 Cross Platform Applications

A single codebase for multiple platforms. That is the idea behind cross platform applications. Usually when a company makes a software intended for a wider user range they need to make an android app, a web app and a desktop app just to get started. This is quite cumbersome since now you have multiple codebases of varying programming languages that needs to be maintained.

Cross platform applications solves this problem by using only a single codebase to represent the business logic. So now the duplication of code can be avoided and only one codebase need to be maintained which is cheaper and more efficient! They achieve this using cross platform application frameworks.



2 Cross Platform Frameworks

Inorder for an app to be truly cross-platform we need a framework which will build to the specific target platforms. Not all frameworks support all platforms. So the framework we choose will depend on the platforms we have in mind. That said there are a few frameworks that are popular and are widely used. They are

- React Native
- Flutter
- UNO



3 Cross-Platform frameworks of today

3.1 React Native

A popular framework created by Facebook in order to make IOS, android apps from a single codebase. It leverages the already widely used and beloved React JS ui library to build ui instead of xml or swift.

React Native consumes native api's using native modules which is made using JSI (Javascript Interface). JSI is used to connect native ui components and api's of the target platform. React Native maps React code to its native component in build time. This way ui will be fast and responsive just like the default implementation.

Since javascript is lightweight, React Native apps have very quick launch times. Also on debug mode, javascript code changes are applied onto the app within a second or two. All these boost developer productivity and fast prototyping.

Disadvantages

- Javascript is notorious for being less secure.
- Immature and some features need native code knowledge to implement.
- App sizes can get pretty large, and it will deter potential users.
- Due to javascript not being type safe, there will be a lot of uncaught bugs during runtime.
- A lot of native api's remain inaccessible as React Native is designed to run on both IOS and Android and most native api's are platform specific.

3.2 Flutter

Developed by Google, Flutter is a cross-platform app framework which runs on Dart programming language. Dart is a relatively new compiled language which is made to render UI.

It's ahead of time compiled nature make the apps made by flutter extremely fast and responsive across different platforms. Also hot reload is almost always less than a second due to DartVM can replace the changed methods and modules directly.

Flutter can target Windows, Linux, IOS, Android and Web. This level of cross-platform support from a single Dart codebase is made possible by using a Skia rendering engine for rendering UI instead of platform specific renderer like other cross-platform frameworks does.

Disadvantages

- Android development can become slow since Flutter uses Gradle as the package manager and Gradle is slow and heavy.
- Cannot access native windows or api's.
- Skia renderer can add overhead and drain more battery in mobile platforms.
- Doesn't provide access to most native api's.
- IOS, Windows and Linux builds are slower.
- First build will be extremely slow.

4 Reference Links

[click to get React Native Research](#)