

Portability in the Enterprise Applications

Anusha V G, Prof. Rashmi R

Abstract— Fast development of applications and its growing reputation in recent years has motivated various IT organizations want to move application between one platforms to another, so portability is a rising concern. Portability is a measure of how easily an application can be moved from one environment to another. For maximum software systems, portability is considered as an attractive quality. Corporate environment such as business or government are designed to operate in a huge software system platform called enterprise application and that uses a pack of technologies that helps them participate directly with customers and are complex, accessible, and component-based, scattered and mission critical. The purpose is to identify some of the portability issues in an application while porting to another platform and considered some key points to achieve the portability using some of the technologies. One of the Technologies is used to overcome the Portability issues in an application that is Full stack Technology where it has different layers based on that there are technologies such as Apache Cordova, REST API, Node.js, Ionic framework and done case study on technology called React Native with real time application.

Keyword: Portability, Enterprise application, Full stack, Apache Cordova, REST API, Node.js, Ionic, React Native

I. INTRODUCTION

Fast development of applications and its growing popularity in recent years has motivated various IT organizations want to move application between one platforms to another, so portability is a rising concern [1]. Portability is a measure of how easily an application can be moved from one environment to another. For maximum software systems, portability is considered an attractive quality. This is particularly true by today's criteria, since there are a lot of various platforms available in all types of software perspectives.

Manuscript received May 22, 2019

Anusha V G, M.Tech., Software Engineering, RV College of Engineering®, Bengaluru -59, India
Email : anushavg.sse17@rvce.edu.in

Prof. Rashmi R, Assistant Professor, Dept. of ISE RV College of Engineering®, Bengaluru -59, India

Conversely, additional planning and resources are hardly ever exactly dedicated to achieving portability, and it is instead most assimilated by AD HOC techniques. Two approaches towards achieving portability between platforms are intermediation and Standardization. Two significant features of portability:

- The transfer from one platform to another should be prospered at the lowest potential effort, cost, and time.
- The capacity to transfer any component of any of the service models across several platforms

It is essential to know the differences between binary portability and source portability:

Binary portability is stated as perfect portability, that is, it is not necessary to change the product in any way for it to work in another environment; it is only required that its executable form is moved. Source portability, it is when changes need to be applied to the source code of the program when porting it, and is preferably measured as degree of portability.

II. QUALITY FEATURE

Portability is a quality feature, so that is divided into six smaller Parts called unit processes. These are defined as the following [2]:

A. Separation

This is the act of sorting out components with exact functionality within the software, which assists to make sure that changes in a definite component affects other components as slight as possible.

B. Abstraction

Abstraction means constructing virtual machines between layers within the software to simplify different conversions and/or match some kind of functionality that is non-native. This is normally more related to separation, since adding separation usually splits the architecture into layers, which might need virtual machines between them.

C. Uniform Composition

Setting limitations on how many the components in the software can be compacted is called uniform composition. This makes it easier to incorporate new components in the system.

D. Compression

Basically the reverse of separation, compression tries to add components together and eliminate layers. It typically serves to raise performance and speed up development.

E. Resource Sharing

Permitting numerous components of the system access to the same resource is called resource sharing. Usual examples of this could be a database or server. This significantly increases the integrity of the system, which in turn supports both changeability and portability.

F. Replication

This is the act of repeating components within a system. Undertaking this serves two resolutions, the first being to improve reliability by having numerous components do the same thing, which lets the system still function even if one or more of the components would flop. The second is to increase performance, by separating the responsibility of performing functionality between numerous components. Inappropriately it is difficult or impossible to accomplish both of these improvements at the same time solely by supporting replication, since they compete with each other.

III. CHARACTERISTICS OF PORTABILITY

The Portability splits into the following [3]:

A. Adaptability

The competency of the program to be adapted to other quantified environments. This could consist of generalizations, parameterizations, layering of components etc.

B. Installability

The ability of the program to be connected in a target environment.

C. Co-Existence

How easily the program can interact with other programs in a common environment.

D. Replaceability

A mixture of adaptability and Installability, it identifies how well a program can exchange another program in a target environment

IV. SOME IDENTIFIED PORTABILITY ISSUES

Recognized the specific points of conflict, which arise while an application is ported from one platform to another and features of an application. Which need to be addressed in a different way in platforms [4].

A. Languages or Frameworks

For each platform has provision for definite language, versions and frameworks. The specific programming languages and frameworks that application has been made will be a key concern while porting it to another platform. For example, languages such as Python and Java, are supported by Google App Engine (GAE) and Java, .NET, PHP, Python and Ruby are supported by Amazon.

B. Platform Definite Services

The time taken for developing application can be really reduced by expending API's (Application Programming Interface). Instead of developing ever functionality from the scratch, they can contribute it into their application by

binding to the specific APIs. A problem occurs when the application essentials to be ported to a various platform there are two cases:

- The application uses full set of services which does not provided by target platform. In these case this need to recreate the lost functionality from scratch on the new target platform.
- The application uses services but provides various APIs in order to use them that are supported by target platform. In these case this need to change the application code and provide it with the APIs of the new target platform. In both cases, these cannot directly be ported across numerous platforms. Needs to be deploy in various platforms the application should be changed.

C. Configuration Files of Platform Specific

Platforms need configuration files in order to initiate the presenting environment and perform application unlike outdated software applications. For example, the "App Engine web.xml" file used by Google App Engine. The way of adapting the configuration files to each target platform enhances to the overall overhead of cross platform distribution of an application.

V. ENTERPRISE APPLICATIONS

Corporate environment such as business or government are designed to operate in a huge software system platform called enterprise application and are composite, manageable, and component based, circulated and mission critical [5]. Enterprise Applications services such as enterprise resource planning and payment processing, electronic billing structures, safety, data management, switching modules, resource planning, manufacturing, application incorporation, collaborative product catalogues, forms automation, sales force mechanization, business intelligence, online shopping, information technology service management and business process management. It also includes news related to information technology fulfilment, office productivity groups, enterprise resource planning, call centres and software as a service.

VI. ARCHITECTURE OF ENTERPRISE APPLICATIONS

These are the various layers of Enterprise application as shown in figure 1[6]:

A. Presentation Layer

It usually user related functionality for management of user interaction with the system, and also involves of service calls for communicating with business logic or application logic through service layer.

B. Service Layer

Consists of service record and information types to correspond with the application logic to divide the application layer as an independent layer. Moreover this layer can be located on various layers, or they may exist in on the same layer.

C. Business Layer

This layer appliance the main functionality of the system, and compresses the applicable business logic. It typically consists of components, among these some of which may be description of service interfaces so that other callers can use.

D. Data Access Layer

It communicates with database to retrieve and add the data in database using its own perspective. This layer expose common interfaces that the components in the application layer can get through.

E. Data Layer

It contains the actual application raw data. Database access can design and maintain this data tier.

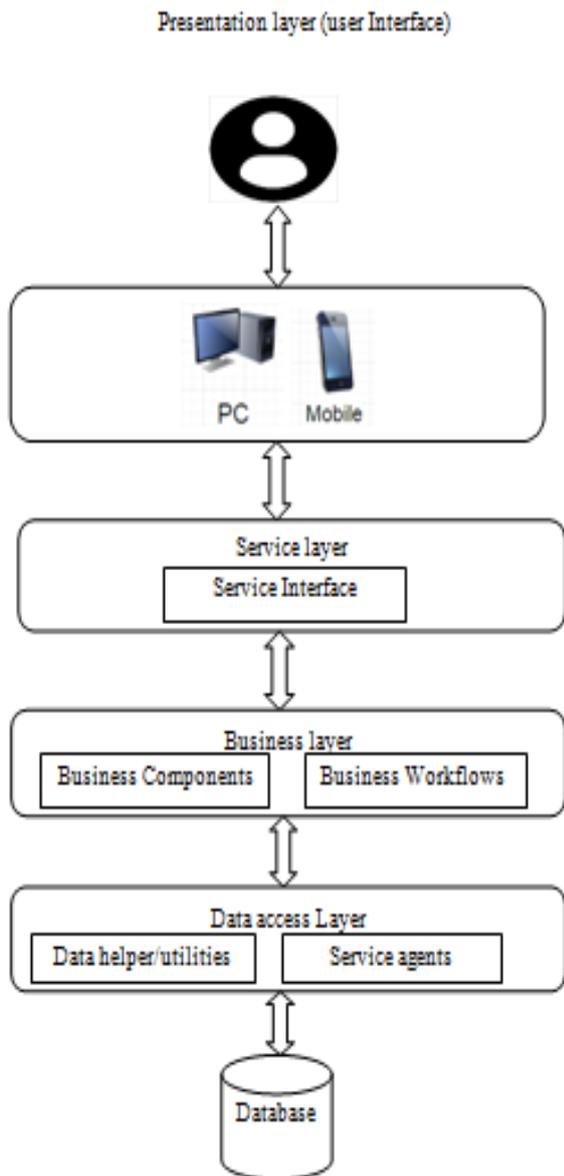


Fig.1: Architecture of Enterprise Application

VII. PROBLEM STATEMENT

The aim of this paper is to understand some of the portability issues and technologies in enterprise applications because in the present world everyone wants to port their applications from one platform to another.

VIII. TECHNOLOGY TO OVERCOME PORTABILITY ISSUES

A. Full Stack Technology

Full stack development is an implementation of both front end and back end portions of an application. This process consists of three layers as show in Figure 2 Presentation layer is a front end portion that act with the user interface, Business Logic Layer or application layer is a back end portion that acts with data validation and the database layer is a data for the application. It looks over of all the process from the commencement of an idea to the real completed product working on each of the various Subsystems of development method proves moderately complex and expensive. Organizations are trying full stack developers who are capable in working across multiple stacks [7].

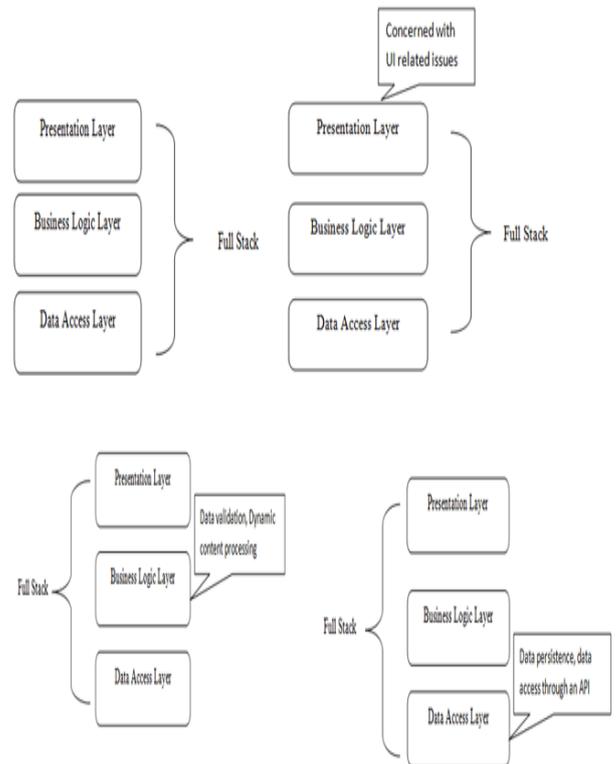


Fig. 2: Layers of Full Stack

B. Features Of Full Stack Technology

Full stack developer is capable of doing as follows:

- Developing progress front end code in HTML, CSS, Java, JavaScript

Portability in the Enterprise Applications

- Developing backend code in Ruby, Python/Java, Aap.Net and using Application Programming Interfaces
- Working with hardware and operating System
- Taking support of Networking, Security
- Performing CRUD of databases
- Managing project and client of organization

Comparison between a full web stack developer and a software engineer:

A full stack web developer: is a person who has knowledge on both front end and back end. They have good programming languages needed for client side development. Know how to create back end applications and APIs that switch the website. They work with databases, operating systems and also handle management of project actions. Basically, they can work on all layers of application.

A software engineer: On the other side, is a computer programmer, who has knowledge of developing applications that runs on the system. They write the code using a language and test it thoroughly to ensure that the software runs without any errors and achieves the required objective. They can work in only one layer of the application.

C. Most Important Technologies

These technologies are under Full Stack to overcome some of the Portability Issues [8]:

- Front end uses the HTML, HTML5, Typescript, JavaScript, CSS, JQuery
- Backend uses Node.js, Angular, AngularJS, Ruby on Rails, PHP
- Database uses MySQL, MongoDB, CouchDB, NoSQL
- Debugging or Version Control uses Xdebug, GIT, Grunt,
- Frameworks uses the Apache Cordova, Ionic, React Native, Frameworks7

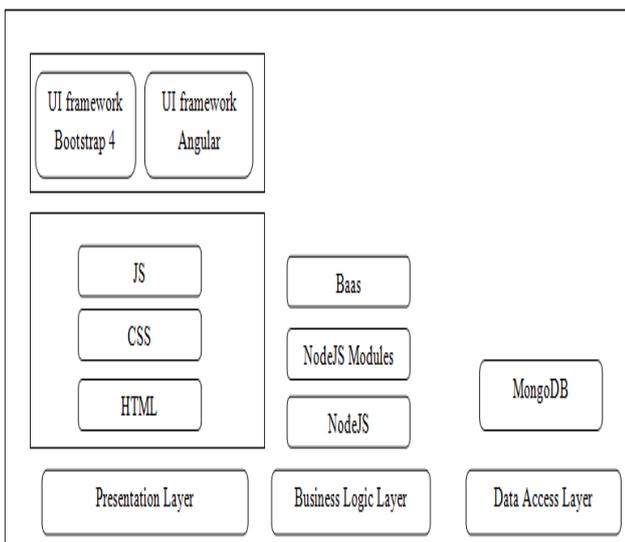


Fig. 3: The full tack Technologies

Figure 3 shows the full stack Technologies used to overcome some of the Portability issues mainly Apache Cordova framework, REST API, NodeJS

D. Apache Cordova

Applications can be ported easily from one platform to another (e.g., Android to iOS or Windows) with the help of a framework like Apache Cordova which is an open-source framework for development. Standard web technologies such as HTML5, CSS3, and JavaScript are used for cross platform development, avoiding each platform native development language. It implements within wrappers directed to each platform, and depend on standards compliant API bindings to each device's sensors, data, and network position.

• Features of Cordova

- **Cordova Command Line Interface (Cordova Cli):** This is a tool that can be used for starting projects, constructing methods for numerous platforms, connecting plugins and set of other useful equipment that make the process easier for development.
- **Core Components of Cordova:** It suggests a position of core components that is requirements of every mobile application. These components can be used for creating basic of the app so that can expend more time for implement of own logic.
- **Plugins of Cordova:** It suggests an API that will be used for developing native mobile roles to the JavaScript app.
- **License of Cordova:** Version 2.0 of Apache License is licensed under Cordova. Trademarks of The Apache Software Foundation are Apache and the Apache feather logos

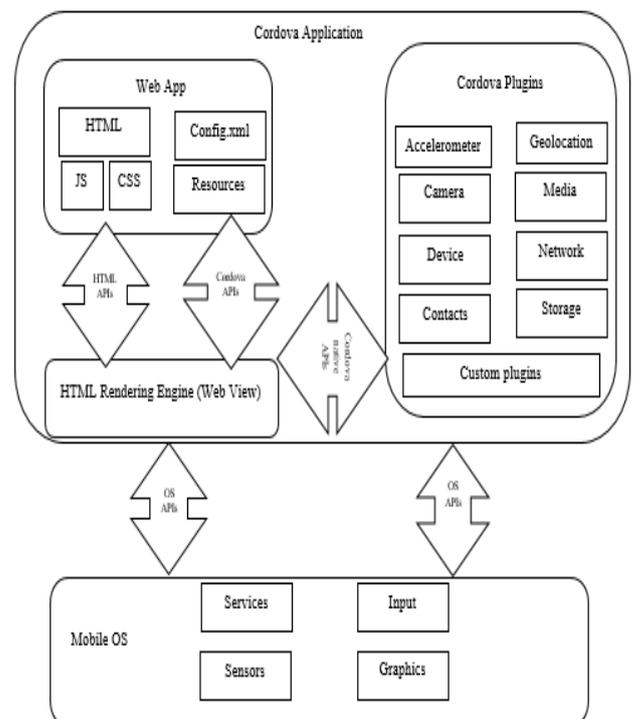


Fig. 4: Apache Cordova Architecture

Cordova also provides the installation of plugins which provides a connection between JavaScript and the native device based on the API as shown in figure 4

E. Rest API

A REST API also called as a REST web service is an application program interface (API) that uses layer on upper of the active service and data access layers supports in resolving two important problems:

- Selection between native and hybrid methods of development
- Code reusability

Here, by using REST API it can standardize the portability from one platform to another. In RESTful architecture, access to resources is just provided by REST Server and REST client access it and changes the resources. Here each resource is recognized by global IDs. Resource like text, JSON, XML of different representations is used by REST. Among that, JSON is the most popular one [9]. Node JS: Node.js is a runtime environment that permits writing back end applications in JavaScript. It is a weightless, capable, and has an ability to run JavaScript on both client side and server side and also opens new opportunities for development. Node.js comes with several APIs appropriate for server side development.

F. Ionic Framework

Ionic framework is an absolute open source framework for development of hybrid mobile app. It provides components, services, and tools using technologies like CSS, HTML, and TYPESCRIPT for developing apps. For Android, version 4.1 and above is supported by ionic whereas for iOS, Version 7 and above.

- Before installing Ionic, Node.js is required
- Based on Cordova Plugins we can develop an ionic application.
- Provides powerful command line interface.

IX. CASE STUDY ON TECHNOLOGY

React native plays a vital role in reducing the development complexity at the same time enhances performance. It is developed by Facebook, which is open source where it can be easily ported from one platform to another platform. It uses only JavaScript, and with the design as that of React and supports third party plug-ins while using less memory and faster loading. It works with all the latest browsers while maintaining the compatibility with the previous ones [10].

X. CHARACTERISTICS OF REACT NATIVE [11]

A. Develop Once and Use It for Various Platform

React Native allows to manufacture state of art Mobile Application in reasonably less time and can be deployed over several platforms, It uses a JavaScript and React to maintain the development and JS engine, hence, learn it once and can develop and deploy applications for many platforms.

B. Identical Look & Feel and Performance

With React Native, developed a legitimate adaptable application that is identical from an application constructed using Objective-C or Java. It utilizes correct iOS and Android UI Components and only set up those UI Components together utilizing JavaScript and React. One of the significant difference is, React Native execute the JS code on the background thread separate from main thread. Animations are affable, as an outcome React Native apps are gracious and responsive.

C. Efficiency

React native is known to be effective in contribution agility, speed and good user experience. It utilizes the React JS framework to provide native feel to the end users. One of the application called Instagram is developed using these technology mainly with the improving performance .before developing using react native there are some of the challenges in it such as Push Notifications view, Edit Profile view, Photos Of view, Post Promote, saving posts, checkpoints because Web Views are good for code sharing and fast iteration speeds, but the UX doesn't feel native and start-up times can be slow. The company overcomes by porting all the views to react native where it has great experience and fast start-up times [12][14].

Table 1: Differences of Application [13]

Applications	Facebook Ads Manager	Instagram	Bloomberg
Problems	The app was facing problems in ad format, date format, time zone and currencies	challenges in it such as Push Notifications view, Edit Profile view, Photos Of view, Post Promote, saving posts, checkpoints because Web Views are good for code sharing and fast iteration speeds, but the UX doesn't feel native and start-up times can be slow	It was facing problem in significant time in developing and updating the different version of IOS and Android app individually
Solutions	By using react native, it has got a good interface, intuitive UX and simple navigation that helps experience to the users	The company overcomes by porting to all views to react native where it has great experience and fast start – ups time	The solution is to adapt For simultaneous updation of app for cross- platforms

XI. CONCLUSION

In today's world, rapid development of application where it can port from one platform to another so Portability is main concern in that sense the portability becomes more important, since the application market is enormously diverse and is a must to stay competitive. Identified some of the portability issues in application while porting to another platform and considered some key points to achieve the portability and done a literature survey on some of the technologies. One of the Technologies is used to overcome the Portability issues in applications that are Full stack Technology where it has different layers based on that three main technologies is used such as Apache Cordova, REST API, Node.js and done case study on technology called React Native with real time application.

REFERENCES

[1] Tom Tisdall, Pankaj Chobharkar and Dae-Kyoo Kim "Challenges in porting enterprise applications to mobile platforms", IEEE, 2018.

[2] Andreas Johansson and Jonas Svensson "Techniques for Software Portability in Mobile Development" IEEE for research and science, 2016.

[3] James D. Mooney "Developing portable software" IRJET, 2014.

[4] Jean Bozman "Cloud Computing: The Need for Portability and Interoperability", IEEE transactions, 2010.

[5] Aparna Vijaya, Neelanarayanan V "Framework for Platform Agnostic Enterprise Application Development Supporting Multiple Clouds" 2nd International Symposium on Big Data and Cloud Computing, 2015.

[6] XIN JIN "Research on the Model of Enterprise Application Integration with Web Services" IEEE transactions, 2017.

[7] Docker "Modern Application Architecture for the Enterprise" whitepaper, 2016.

[8] Qin Li Michael "Understanding the Performance of Enterprise Applications", IEEE transactions, 2016.

[9] Martin Hafner Robert Winter "Processes for Enterprise Application Architecture Management", IEEE transactions, 2017.

[10] Lvqing Yang Chuan Jiang "Research of Enterprise Application Software Project Implementation Model", IEEE transactions, 2016.

[11] Magdalena Kostoska, Marjan Gusev, Sasko Ristov, "a New Cloud Services Portability Platform", 24th DAAAM International Symposium on Intelligent Manufacturing and Automation, 2017.

[12] Fotis Gonidis, Iraklis Paraskakis, Anthony J. H. Simons, Dimitrios Kourtesis, "Cloud Application Portability: An Initial View", Balkan Conference in Informatics, BCI '13, Thessaloniki, 2016.

[13] D.Petcu, "Portability and interoperability between clouds: challenges and case study", in Proceedings of the 4th European conference on towards a service-based internet, Poznan, 2015.

[14] D. Petcu, G. Macariu, S. Panica, and C. Craciun, "Portable Cloud applications: from theory to practice", Future Generation Computer Systems, 2016.