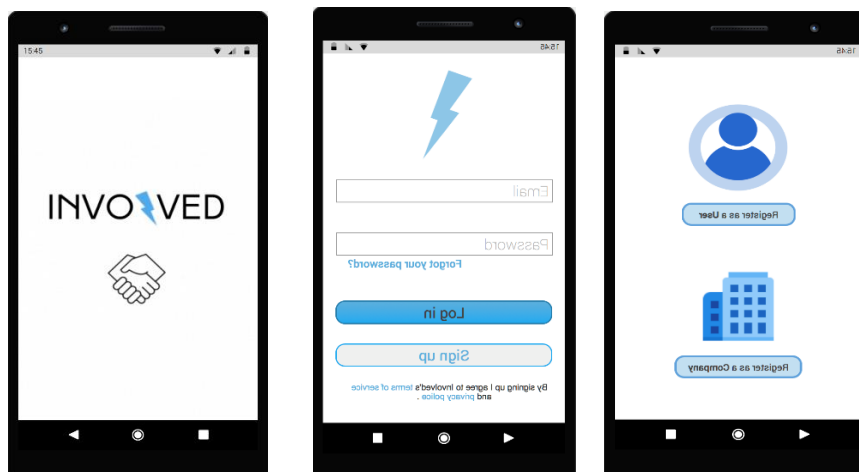




MISSION

INVOLVED's mission is directly correlated with all those initiatives that aim to create a more sustainable environment within cities. It is a platform that emphasizes on matching companies with individuals, that want to drastically reshape physical spaces within cities by proposing and implementing innovative initiatives. Everyone can download the application for free and start participating in events that take place constantly from well know organizations all over the world. The platform also gives the ability to those who want to share their achievements the opportunity to show case certificates of participation depending on the event they attended. Those certificates are accredited by our platform and are completely shareable through social media and other platforms as well. Our goal is to motivate as many people as possible to be part of our culture and mission, to achieve the change we vision.

APP FRONT-END INTERFACES



15:45

What's your name?

First Name

Last Name

← →

15:45

Your Email

Email address

← →

15:45

Your Phone Number

Number

← →

15:45

Your Address

Country

City

Address line

Postal code

Number

← →

15:45

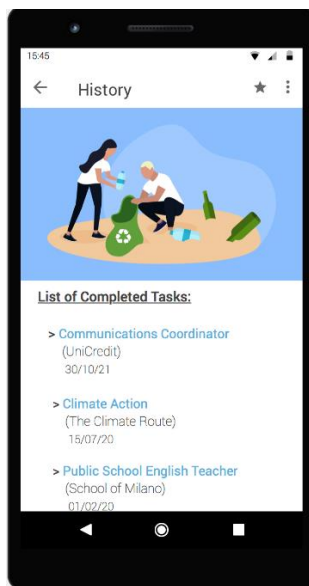
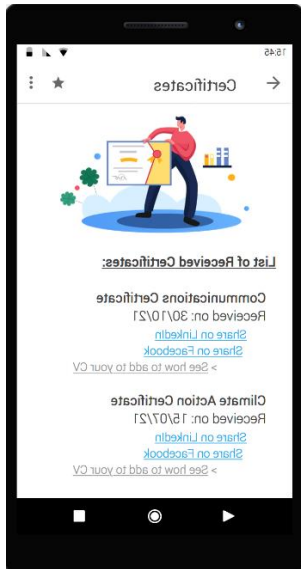
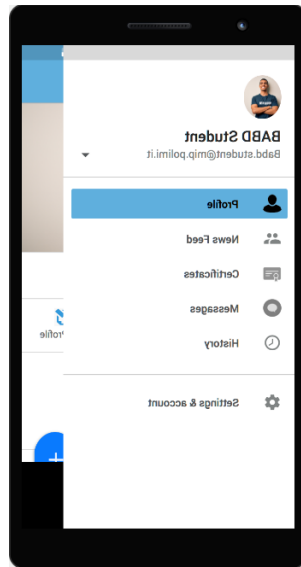
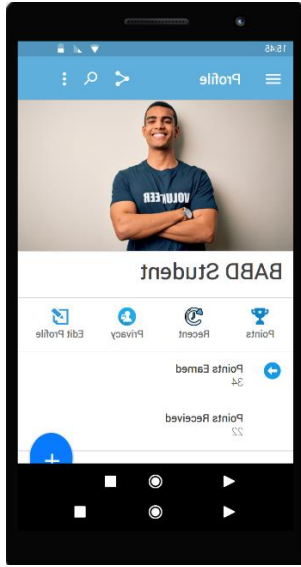
Create a password

Password

Confirm Password

← →





BUSINESS MODEL

The application will operate as a multi-sided platform, providing network benefits to the two sides of our customer base. The one being companies, organizations and institutions of all sizes wanting to promote initiatives and organize activities to further transform communities into more sustainable environments for future generations. While this is happening, the other side will consist of ambitious individuals; users who want to contribute to community and want to connect with an organization that shares the same passion for change as they do.

The platform will provide users with a reward system to update their profile status every time they take on an activity or participate in an event. After collecting a significant amount of points and reaching achievement milestones, individuals will be given participation honor certificates, that can be completely shareable through other platforms for everyone to share their work and motivate others to do the same. The app will provide its services for free to the most price sensitive side (end-users), while companies and organizations will have to pay a monthly fee for having access to the platforms network.

APP MOBILE ARCHITECTURE

Mobile app architecture refers to a set of rules, techniques, processes, and patterns to develop a mobile application. These rules help developers create an app that meets both the business requirements as well as industry standards.

Mobile app architecture is made up of all the parts of the app – all the questions about why, what, how – including what data is collected, how the data moves, what the app looks like, for what platform, using what tech stack.

Lack of architecture results in an app:

- Longer and more costly to develop
- Hard to maintain, particularly if staff change
- Harder to build upon or scale
- Difficult to test
- More prone to errors

On the other side a good mobile application architecture will enforce good software development principles (KISS, DRY, SOLID) in the appropriate stages of development to help accelerate development, providing a clear path for data flow that makes work easier and supports clarity over how to scale or expand the app in the future.

A clearly defined mobile app architecture helps to support flexibility and Agile development methods, makes testing more efficient and makes future maintenance easier and less prone to bugs. A strong mobile app architecture will save both time and money in the short and long term.

A good architecture will not be platform specific, but rather apply to native and cross-platform choices, resulting in a unified approach to development. If we think of

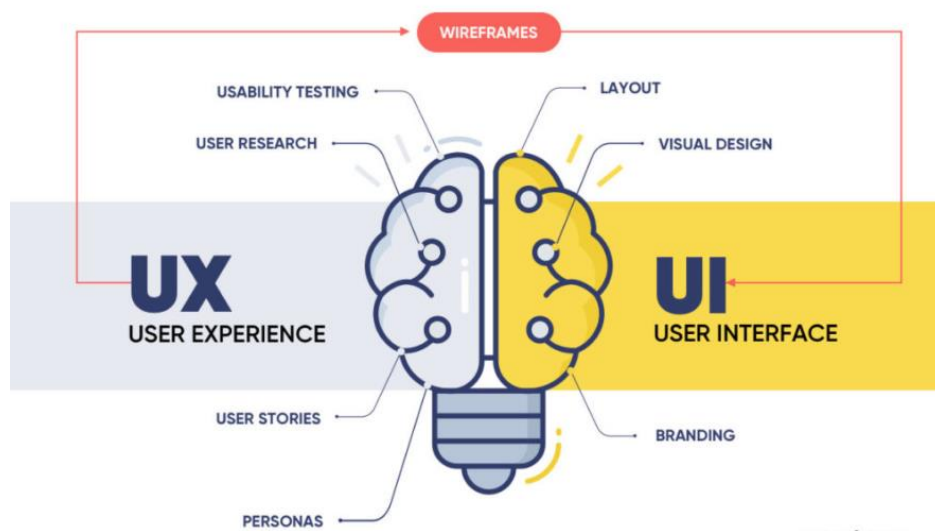
mobile app architecture as a skeleton for how we create a mobile app, we can then define layers of how we build out the vital components of the app.

The most popular representation of mobile app architecture is represented by 3 layers: Presentation, Business Logic, and Data.

The presentation layer consists of all the processes and the components to deliver the app to the user. When building the presentation layer, developers are concerned with what the user sees and feels when using the app. In other terms, the presentation layer is made up of the UI (user interface) and UX (user experience).

User Interface is concerned with design questions such as colors, fonts, placement, and overall design.

User Experience manages the way a customer interacts with the app through a detailed understanding of what a user wants and feels



The Business Layer is concerned with the logic and rules responsible for data exchange, operations, and workflow regulation. This layer is responsible for:

- Security
- Data caching
- Logging
- Data Validation
- Exception management

The business layer can exist on the server or on the user device, depending on the operations of the app and the resources each operation takes.

The Data Layer includes all the data utilities, service agents, and data access components to support data transactions. This layer can be thought of in two parts:

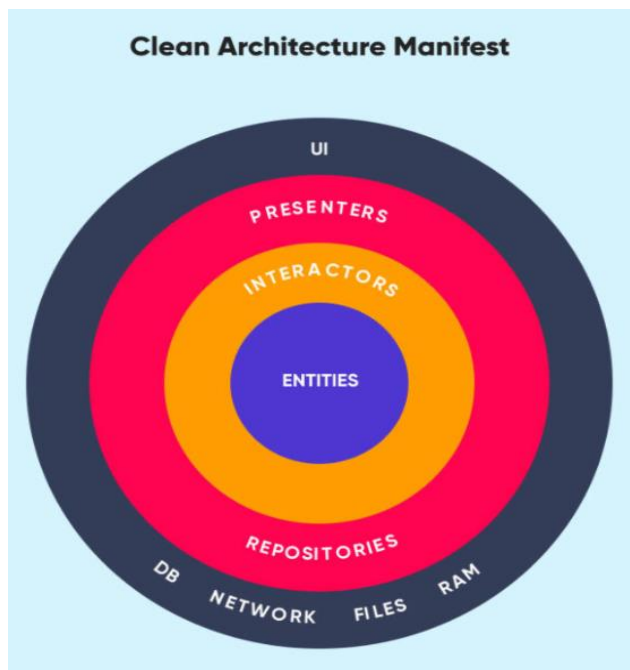
Persistence: Data access with data sources via API

Network: Network communication, routing, error reporting
The design of the data layer must include thoughts about validation and maintenance of the data.

ANDROID MOBILE APPLICATION ARCHITECTURE

Apps developed specifically for Android are one kind of native app – an app developed for a specific mobile platform. Android apps are developed to support Android languages (Kotlin and Java) for devices from a variety of manufactures including Google, Samsung, Sony, and Nokia. There is not a single architecture recommended for Android, but the most accepted Android architecture for mobile apps is Clean Architecture.

In Clean, the architecture is built on the principles of layers and inversion of control. Clean focuses on the same 3 layers structure modeled above, with the business layer sometimes referred to as the domain layer. In Clean architecture, the domain/business layer must not depend on the other layers, but instead must leverage interfaces. While this can be difficult to understand, it does make it easy to add to and scale apps over time.



IMPORTANTFACTORS TO CONSIDER WHEN DEVELOPING MOBILE APP ARCHITECTURE

1. DEVICE TYPE

When designing a mobile app, first choose the platform (IOS, ANDROID, WINDOWS, CROSS-PLATFROM), then consider the different models of smartphone that are in use. These are important pieces of information to help determine the ideal dimensions for development.

The following factors are important to consider when designing a mobile app:

Screen size and DPI

Screen resolution

CPU (processor)

RAM (memory)

The goal in this stage is trying to deliver the most consistent experience across various platforms and device sizes (tablet vs mobile, model varieties) so that every user – regardless of their device choice – has the best possible experience.

2. DEVELOPMENTAL FRAMEWORKS

Developmental frameworks are a consideration in designing the mobile app architecture as well as in establishing the tech stack. Frameworks provide libraries and basic templates and components for building web apps, both for front and the back- end.

Front-end frameworks for building mobile apps include Bootstrap, Foundation, React, Angular, Vue, and Backbone. On the back-end (server-side) developmental frameworks depend on the chosen programming language and target platforms, including Ruby on Rails, Flask, Django, Laravel, Swift, Xamarin, React Native and Flutter – among many others.

3. Bandwidth Scenarios

User research is important to understand more about the target user. Around the world, users experience different bandwidth limitations, with some countries on 5G and others still experiencing spotty connections. A highly interactive, graphic-heavy app will not be appropriate for apps that target rural users, for example.

4. USER INTERFACE/ USER EXPERIENCE DESIGN (UI/UX DESIGN)

When it comes to mobile app architecture, design plays an important role in first impressions (how it looks – a strong UI) as well as keeping users around (how it works- a strong UX). A strong UX design is critical to online success, but mobile UX is complicated by changing user expectations and best practices of each operating system (OS) and device type.

The mobile app architecture must balance UI against UX in the design phase. Start by understanding the basics of mobile UX design and the latest mobile UX design trends for 2021 to ensure the app is delivering value.

5. NAVIGATION

Navigation is the users direct contact with the design, impacting both the front-end and the back end. A great mobile UX design helps users easily identify how to move around the page and explore further sections. Familiar it is key to navigation. The following navigation best practices help ensure the mobile app is easy to use, reducing friction in the customer journey:

Hamburger menu: The three-line menu in the navigation drawer or top bar is popular with mobile apps because of its familiarity. Enhance navigation with a hover over menu, reducing clicks.

Search: A well-positioned search bar levels up usability, with standard position being the top right

Bars, rails, drawers, or tabs: There are many options to navigate around an app, including fixed bars of buttons (top and/or bottom), rails (a vertical bar), drawers (hidden navigation), and tabs (screened content with fixed titles).

Familiar icons: Familiar icons such as home, search, photos, folder, etc. make navigation easier

Intuitive labeling: Helping distinguished information with labels that spell out the intent of a button, option, or feature.

Site organization: categorization can make site navigation easier – or more complicated – depending on how the categories are named and organized. Defer to information architecture best practices.

Gestures – supporting gesture-based navigation (swipes) can streamline navigation.

Scrolling – how the app supports scroll vs fixed elements.

Thumb zone navigation – Design with the thumb zone in mind for enhanced usability.



The best way to ensure positive user experience (UX) with navigation is through research and user feedback. Wireframes are an important part of early testing of what the app looks and how it can be navigated.

6. PUSH NOTIFICATION VS REAL-TIME UPDATES

There is a careful balance to be had between nudging users and annoying them when it comes to notification frequency and method. Notification frequency can also have an impact on device battery life, which can in turn impact user retention.

Mobile push notifications are messages that apps send to inform users they have a new in-app message, either from the app itself (such as product updates, offers, or reminders) or from another user. Android users are automatically opted in for push notifications, but iOS users must opt-in, and typically do so at a lower rate (51% iOS vs 81% Android).

While some may consider them annoying, push notifications can be a great marketing tool. Push marketing is most effective when it is personalized or offers a sense of urgency.

Moving past scheduled push notifications, mobile apps are leveraging real-time technologies to add greater interactivity to their apps. For example, a delivery app can use real-time updates to provide updates about when a food order has been accepted, prepared, picked-up, or tracked on a live map. Other examples include health and fitness apps or home technologies that benefit from real-time alerts (such as security systems). Like push notifications, real-time updates have their time and place in order to avoid overwhelming users with notifications.



HOW TO CHOOSE THE RIGHT MOBILE APP ARCHITECTURE?

- **Budget**

The choice of architecture depends on developer skill set, market analysis, and development approach. Unfortunately, the need for rapid, agile development often outpaces availability of IT skills for native and cross-platform developers – time and delay which can make an app obsolete by the time it hits the market.

- **Audience Analysis**

Identify users, their personas (backgrounds, needs, and goals), competitive analysis, agile user stories, flows, mapping, wireframes, and prototyping tests.

- **Key Feature Requirements**

Let the business requirements drive the feature lists, determining whether the app is best suited for Native, Hybrid, or Cross-Platform development.

- **Platform Choice**

The best practices around UI and UX will vary depending on the platform.

- **Development Time**

Some architectures require more time to develop certain elements or integrations, which should be built into the overall plan and choice.

- **Start Developing A Test App**

Agile development focuses on iterative development and continuous feedback, focusing on customer-centric input that can only come from a test app. This very basic app contains basic functionality and navigation and provides valuable feedback as to whether the architecture is working to satisfy the target user.

CLIENT SERVER (CS) MODEL

Application architectures are often modeled to highlight or illustrate the overall layout of the software (application code and platform) and hardware (client, server, and network devices). While there are many possible combinations of software and hardware, application architectures often fall into a series of recognizable patterns. Application architectures are commonly modeled in terms of a client- several architectures were in one or more client devices requests information from a server device. The server typically responds with the requested information. We can further consider client-server architectures using layers and tiers and the communication between the layers and tiers.

LAYERS

Application code functionality is not necessarily uniform throughout an application. Certain sections of application code are better suited for handling the user interface, while other sections are developed to manage the business logic or communicate with the database or back- end systems.

Layering describes the division of labor within the application code on a single machine. Layers are often no more than code modules placed in different folders or directories on the client or server.

With client – side code, there are generally zero to three layers of application code. With server-side code, there are generally one to three layers of application code. This is partly a matter of good software design that helps code re-usability, partly a matter of security, and partly a matter of convenience.

A client with zero code layers essentially has no custom application code. This type of client is commonly referred to as a thin client and is possible in client-server architecture if the server holds all the custom application code. A client with one to three layers of application code is commonly referred to as a fat client.

A server can also have one to three layers of custom application code. However, you cannot have zero code layers on a server.

The code layer that interacts most closely with the user is often referred to as the Presentation Layer. The second layer is often referred to as the Business Layer, as it typically handles the business logic of the code. The third layers often referred to as the Data Access Layer. It typically handles communication with the database or data source.

TIERS

While breaking up application code functionality into layers helps code re-usability, it does not automatically make the architecture scalable. To do so, it is important to distribute the code over multiple machines.

Tiers describes the division of labor of application code on multiple machines. Tiering generally involves placing code modules on different machines in a distributed server environment. If the application code is already in layers, this makes tiering a much simpler process.

The code that interacts most closely with the user is often placed in the Presentation Tier. A second tier, which holds the application business logic and data access logic, is often referred to as the Application Tier. The third tier often houses the database or

data source itself and is often referred to as the Database Tier. This is an example of a three-tiered architecture.

The servers that make up each tier may differ both in capability and number. For example, in a large-scale distributed web application environment, there may be many inexpensive web servers in the Presentation Tier, a smaller number of application servers in the Application Tier, and two expensive clustered database servers in the Database Tier. The ability to add more servers is often referred to as *horizontal scaling* or *scaling out*. The ability to add more powerful servers is often referred to as *vertical scaling* or *scaling up*. Tiering the application code in such a fashion greatly facilitates the ability to scale applications.

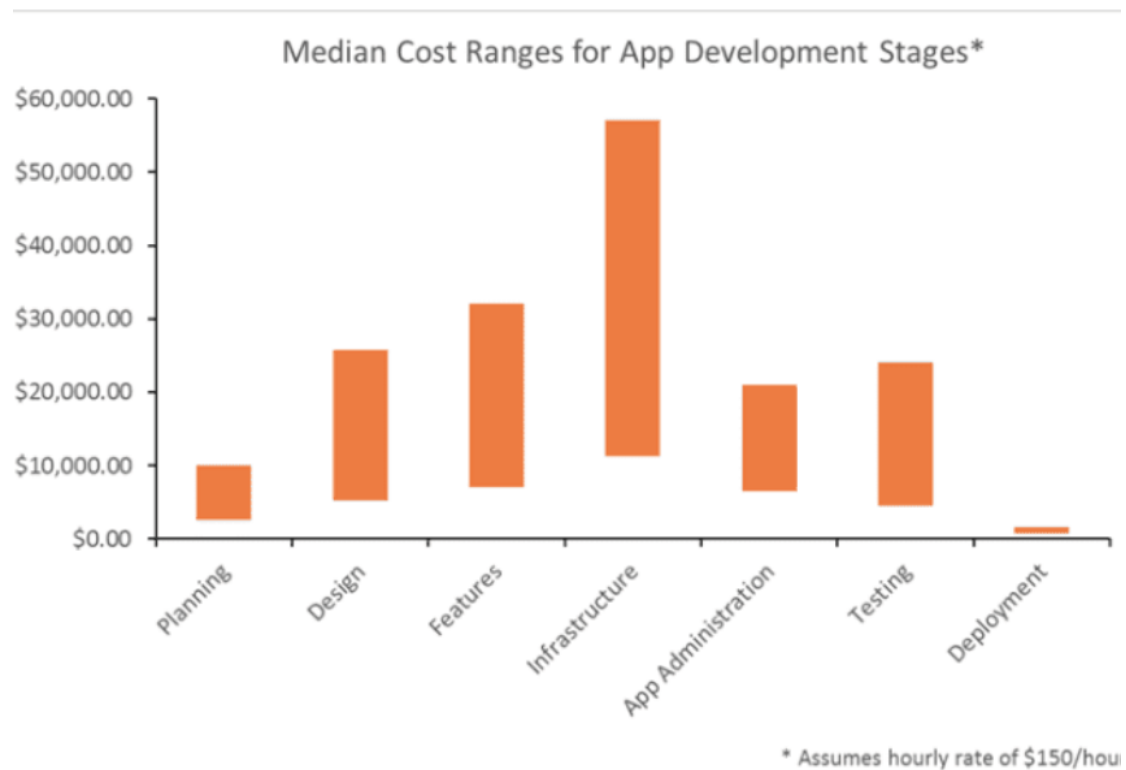
In large-scale distributed web applications, tiers are often bounded by firewalls. For example, a firewall may be placed in front of the Presentation Tier while a second firewall may be placed in front of the Application Tier. The Presentation Tier is thus sandwiched between firewalls in what is termed the Demilitarized Zone (*DMZ*), while the Application and Database Tier servers are shielded behind the second firewall in what is termed the Intranet Zone. Tiering therefore also facilitates security and allows large enterprises to shield precious internal systems from traffic originating from untrusted zones such as the Internet and *DMZ*. Without tiering, it becomes very difficult to secure internal systems.

Tiers generally describe server architectures, and we do not typically count client devices as a tier. While it is possible to do so, this is not a usual convention.

APP FEATURES

Medium Range Complexity App for Android

- **Completely customized UI features** (Custom UI provides a way to build the user interface of your app. Using custom UI, you can define your own user interface using static resources, such as HTML, CSS, JavaScript, and images)
- **Accepts payments**
- **Has API integration**
- **Adaptable for headsets, tablets and computers or other digital devices**
- **Back-end server**
- **Roughly 47,000 - 80000 \$ to build**



TO SCALE OUR APP QUICK AND EFFECTIVE

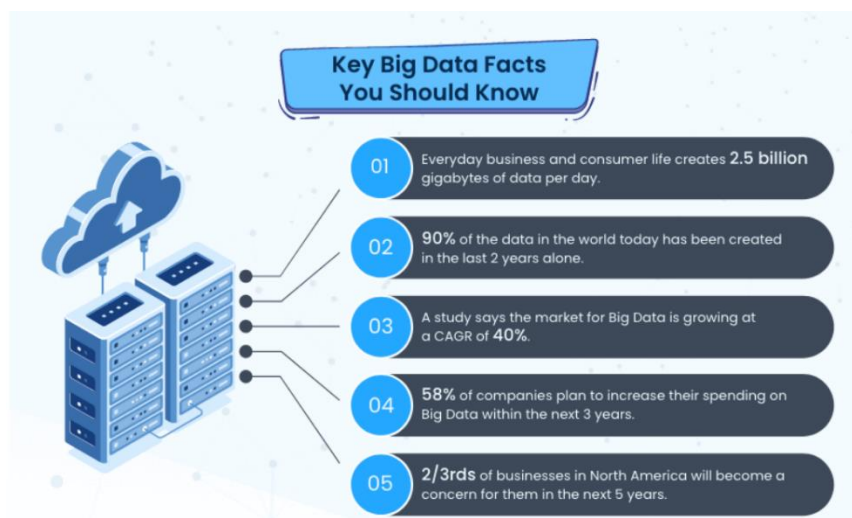
KPI Targets to consider:

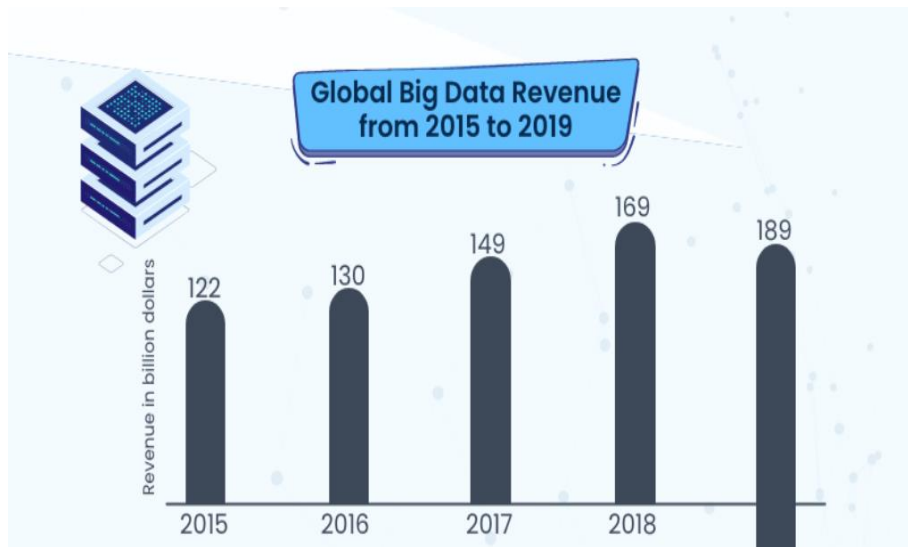
- 1) **Average monthly organic growth** (10-20% is considered a good benchmark)
- 2) **D-1, D-7 (14%) and D-30 Retention** (are calculated as the percentage of unique users who are active at any time after 1day, 7 days and 30 days of signing up or installing the app)
- 3) **LTV (lifetime value)** is an estimate of the average revenue that a customer will generate throughout their lifespan as a customer
- 4) **Target MAUs (Monthly active users)** measures the number of unique users to your app over a 30-day period
- 5) **IAP In App Conversion Rates** refers to the percentage of users who complete an in-app purchase within an otherwise free service or app. For example if you had 100 people using your free app and 3 of them made some sort of purchase within the app, you IAP conversion rate would be 3%.

HOW BIG DATA CAN IMPLEMENTED TO APP DEVELOPMENT + HOW TO ACHIEVE A DATA DRIVEN BUSINESS MODEL

Data is the most important raw material for an online business. Nowadays, an average user has higher expectations than ever — as developers get better at fulfilling their requests, they demand more faster and better services and software.

Big data is data, that grows exponentially as time progresses and it is immensely complex. Hence, it cannot be managed by conventional tools that handle regular data.





All mobile apps need data to function, however the main problem that all apps are facing is that they must deal with a lot of raw data in an unstructured form. Big Data needs a degree of high-level analytics for discovering the figures and developing insights to benefit a mobile app business.

In the development phase, for market research purposes, collecting necessary data from your app users and using an analysis software to analyze them is a good way to identify market trends and customer preferences allowing you to build unique features.

Smoother Analytics Process: Big Data can help app creators to understand the behavior of app users. This analysis can help app creators to optimize their apps and improve the user experience. Mobile app developers can take these behaviors as cues to improve and innovate their future apps, by adding extra features that result from customer behavior analysis or feedback.

Better overall App performance: Big Data analytics allows developers to get a better insight of their app traffic. This allows them to track individual segments and observe the data for glitches. They can fix these issues and boost the overall performance of their app.

Easier to Scale Up: Big Data can be used for marketing purposes, helping app developers to gain insight on their customer base regarding their needs, wants, preferences and pain points etc. This data can be used to boost lead generation and conversion rates. On the other hand, this can also help an App to scale up user engagement and revenue at the same time, especially if those metrics are highly positive correlated.

Real Time Exploratory Data Analysis (EDA): The biggest advantage of Big Data development is that all data are being generated in real-time. What that means practically, is that whenever the management wants to initiate the launch of a new digital business strategy, the team of analysts in the organization can perform EDA on updated customer data. That is very important since it provides accuracy, reliability, low risks, and up to date insights on the feasibility of new business proposals.

Easier to Reach a Global Customer Base: The main purpose of Big Data is to analyze huge data at fast speed. This is exactly why big data analysis is perfect for apps that

reach a global customer base. This will help companies to reach a larger customer base and ensure better overall conversions and revenues.

More Efficient Marketing: Market research through big data is already a rising industry. Marketing researchers are mostly collecting Primary Data to answer specific research questions like “What new features do our customers want added to our product? “, “What customer segments are the most attractive target groups for our services?”, “How should we position our proposal to distinguish ourselves from competition? “. Primary data is expensive to obtain, because its data retrieved randomly and independently from a population to be statistically significant.

Big data analytics are “technologies (e.g., database and data mining tools) and techniques (e.g., analytical methods) that a company can employ to analyze large-scale, complex data for various applications intended to augment firm performance in various dimensions”

How Big Data Works

Data Collection This is primarily done through various applications like Instagram, Facebook, twitter, YouTube, and phone calls in the first phases of market research, mainly before launching the App. Afterwards customer data can be collected mainly by monitoring in-site traffic, subscriptions, and registration data; after customers have agreed to a consent policy that will allow the App developers to have access to their data and their on-site behavior while using the App.

Integration After data collection, data is processed for making sure that the business analyst don't have major technical difficulties using it. 80% of collected data is projected to be categorized as unstructured data, but this should be of no surprise considering the number of photos, videos, documents, and emails generated in the web by the minute. Hadoop tech can be used to integrate this kind of data. Unlike structured data that comes neatly organized in relational databases, unstructured data can be classified in to 2 different groups:

Logs: This group consists of application logs, stored as files that list events such as page visits, button clicks, logins, exceptions, and so forth. Part of this log lines can be structured to contain the date, log type and URL, while the other part can remain unstructured with any info the app's developer choose to include.

So, let's say that we need to come up with insight on – which transactions happened, how long they took and what (if any errors) occurred. The only way to manage such tasks is to write custom code (using Hive or Pig), but still, we should find or write UDFs (User Defined Functions).

Fully Unstructured Data: Data such as social network statuses, emails, documents, images, and videos. Basically, their content is fully unstructured and may appear in form of free text or binary bits and bytes.

Processing this data means extracting structure from it, with the use of advanced algorithms.

To manage all these, it is important to leverage technologies from companies that license their software, to have access to high-power, cloud-based solutions, visualized data guidelines for automated data flows – that can allow us to transform, normalize and clean data.

Data Handling Data should be stored and handled in a way that will allow to be assessed easily. Big data storage can handle very large amounts of data and ensure their encryption and protection from cyber-attacks.

There are currently two well- established big data storage methods:

Warehouse Storage: Is a large building facility which its primary function is to store and process data on an enterprise level. These warehouses are usually optimized to retain and process large amounts of data at all times while feeding them in and out through online servers where users can access their data without delay. The greatest benefit of data warehouses is the ability to translate raw data into information and insight.

Hadoop is a software framework meant for distributed storage and processing of big data to handle massive amounts of data and computation.

Cloud Storage: With cloud storage, data and information are stored electronically online where it can be accessed from anywhere, negating the need for direct attached access to a hard drive or computer.

The cloud also provides the ability to scale this infrastructure quickly to manage large increases in traffic or usage.

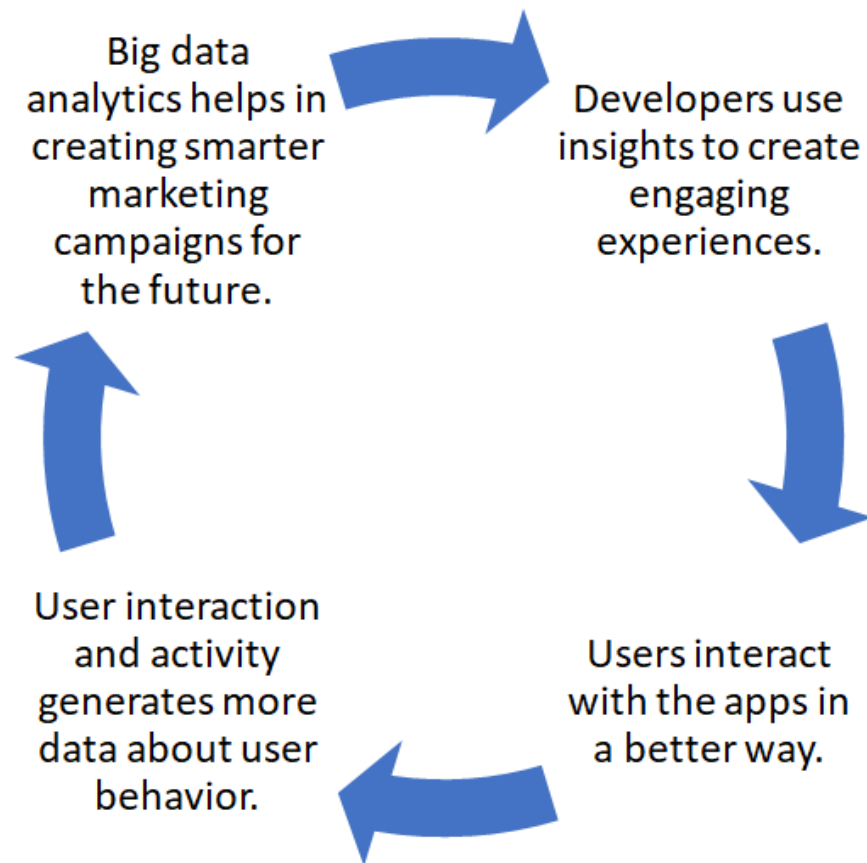
Cloud storage is also significantly cheaper than physical storage of data. Data warehouses consume large amounts of power, space, resources and come with more risk.

Analysis

The role of big data in mobile app development is not only for assessing the data and providing your insight, but it runs more intensely.

Analysis of User-Experience

Aside from customer requirements, developers are susceptible to know the ways users utilize their mobile apps. Hence, using big data app development, you can conduct a detailed analysis of user experience. As an outcome, it provides a detailed analysis of user engagement for every feature and page. You can utilize similar data for preparing a list of everything that users require, want to change, or enhance.



Customized Marketing Campaigns

User behavior data analysis incorporating requirements, likes, dislikes, and expectations can create customized marketing campaigns. With big data, you can assess buying patterns, demographic data, and users' social behavior for changing your marketing plans as per their present requirements. By creating the right tactics, you can fuel engagement, drive adoption, build app revenue, and boost satisfaction.

Sales Conversions

Targeting constant updates for the App, to introduce further features that will meet customers need from both sides of the transaction, will lead to accelerate the growth of the customer base from both sides. That practically means that total revenues will increase, since this is the way a two-sided business model operates. EDA on easily assessable, structured bid data is the center stone of this digital business strategy.

Connected Devices

IoT is a superb way of moving ahead using automation and develop in the market with the help of the right tech use of big data.

IoT helps ease complicated procedures and connect gadgets with apps for users. So, what is the role of IoT devices and big data in app development? IoT devices help

gather data from users and assess them for actionable awareness. These help developers get user-friendly and result-oriented applications that can influence the market.

HOW TO IMPROVE OPERATIONS

- 1) Use Big Data technology for predictive analytics, by implementing unsupervised machine learning classification models to already transformed data frames that are distinguished to train/test data. This will allow us to post relevant events on each user's main page, that are going to be related to previous searched items.
- 2) Real – time data management will help our business plan to remain flexible and agile during periods of constant turbulence. By being able to analyze changes in real time we will be able to adjust and target the most sensitive audience side of our transactions.
- 3) With IoT and Cloud Technologies we will be able to scale our operations from an initial niche market to a global one quick. By implementing data visualization tools to draw correlations from customer characteristics from different target audiences and competitors will allow us to add extra features to grow our customer base from both sides; while at the same time evaluate which of those metrics affect sales and revenues.
- 4) Data collection on customer's actions, behavior and characteristics can be further managed to be sold in 3rd parties for advertising purposes. This is a business strategy that comes in later stages of the project plan, but it is a revenue source that cannot be dismissed.
- 5) Use data to build a loyalty program to increase customer retention will help to reduce costs and achieve business growth. A point reward system, that will increase customer engagement with our app.

This whole procedure will start by giving some initial “free” points to new users that just registered on the app. Afterwards users will be awarded with points based on the nature of the event they will be attending and the amount of their contribution. This reward system will be announced by each company that promotes the event in collaboration with the reward policies implemented by the App management. As user collect more and more points by contributing to different community activities, he will be awarded with participation certificates that he/ she will be able to share on social media and other platforms.

Users will also be rewarded with points for providing feedback regarding their satisfaction level based on our platform services and by stating their opinion on areas that the app needs to be improved. Creating a more personalized connection with the users will be the key for a fast scale up to global standards.

To keep track with all this information and data collected, all requirements mentioned above regarding a structured Big Data management strategy, should be in the core of our operations.

