KLEF Play

Meghana Komatineni[1], Manaswini.K[2], Nikhila Medikonda[3], M Devendra Sai[4], G. Pradeepini[5]

[1-4]Student, *Department of CSE, Koneru Lakshmaiah Education Foundation, Vaddeswaram, Andhra Pradesh, India.*
[5]*Professor, Department of CSE, Koneru Lakshmaiah Education Foundation, Vaddeswaram, Andhra Pradesh, India.*
---***--

Abstract- *KLEF-play could be a video streaming progressive internet application that uses YouTube as a server and storage for videos and therefore the app itself resides on our own desired server. KLEF-play achieves this by using embed features provided by YouTube and thus will even be hosted on a smaller server while not carrying the load of videos. It's developed to figure on any given device and any given platform provided, the given device contains a browser. Associate in app shell design is enforced in KLEF-play with the employment of service staff and JavaScript for the technology stack. It uses React.js for developing user interfaces and achieving one page application model for the additional native-like expertise. Node.js and MongoDB are those powering the rear finish back end of the application. This paper primarily focuses on the design and dealing of the applying and our main focus is to gather all necessary academic videos of faculty together with lectures, tutorials on one platform with correct content management as desired by the institute or organization using it.*

Key Words: Embed, Progressive Web App, React JS, Node JS, MongoDB, Online Video Streaming, Hybrid App, Video Encoding.

1. Introduction

In the world of 4.5 billion net users and astounding net speeds, it's straightforward for folks to access the net whenever they require and for no matter they have, particularly videos. This has enforced most of the activities being carried on-line together with learning. Websites like Udemy, Coursera, EDX and YouTube are already on the highest and are being employed by several professionals. The foremost viewed platform amongst all is YouTube and permits every kind of video content to be uploaded on that whereas the previous ones don't and that they work on a particular niche. **[1]** However thanks to scattered content on the net, the power to form the audience specialise in what any given establishment needs lacks. Self-hosting videos could be a dangerous plan as streaming video wants loads of transcoding and serving them in keeping with the client's device specifications. Users largely access the net through their smartphones associate degreed thus an application should load quickly and work quick. **[5]**

KLEF play could be a video streaming, progressive internet application for such establishments. It uses YouTube to transfer videos and embeds them on KLEF play. YouTube servers are best at transcoding videos and providing them at their very best quality and formats in keeping with the client's device specifications and net speed. KLEF play whereas provides a lot of management over what users will see and access. Videos are set to unlisted on YouTube in order that they're solely visible on KLEF play.**[1] [2]** Inside smaller organizations, it will play a significant role as a targeted social media with a lot of management of everything that is uploaded on KLEF play.

A straightforward approval system is enforced for verification of content uploaded on the appliance. A gaggle of admins is appointed to manage the content and users on the appliance and thus becomes a superbly managed media app for establishments. KLEF play is enforced exploitation of Javascript. It uses React.js to accomplish one page application model and simple state management with insane speeds. The front-end works while not reloading the net page and renders parts rather than hypertext markup language pages for every request sent. This avoids causing multiple HTTP requests to the server and saves loads of your time and makes the app quicker. **[4]**

Rear implements an are composed of MongoDB and Node js, once more Javascript. NoSQL information like MongoDB is employed as a result of it's loads quicker, easier to implement with node and versatile for future functions. KLEF play could be a progressive internet application or PWA, whereby progressive means that for every kind of user. This makes KLEF play run on any given device and any given specification, condition, the device includes a browser. PWAs implement multiple trendy techniques like caching to produce quicker and fascinating apps. **[3]**

2. Literature Survey

The Goal is to make a progressive video streaming net application that uses YouTube as the server and storage for the videos and also the app itself is on the server we would like.

Here Methodology we tend to area unit victimization is by primarily specializing in the design and operation of the appliance and to gather all the necessary instructional videos of the faculty as well as lectures, tutorials in an exceedingly single platform with content management as desired by the institute or the organization that uses it. **[1]** KLEF-play is enforced victimization of Javascript for full-stack development. The app leverages react.js for user interfaces. React provides higher ways of changing the DOM or document object model as a result of it solely changing a part of the DOM that has been updated.

Here within the alternative paper **"Video Streaming: ideas, Algorithms, and Systems"**, Video streaming is to cut back unconnectedness within the video signal, that is, to cipher solely the video options that area unit perceptually necessary and not waste valuable bits on info that's not perceptually necessary or inapplicable. **[2]** Wherever the advantages of this paper area unit low delay before viewing starts and low storage necessities since solely a little portion of the video is kept at the consumer at any purpose in time.**[2]**

Here in an Paper "**Splay: A Lightweight Video Streaming Application"**, the advantages of this one are may be a strong and reliable resolution for small-scale and large-scale establishments and organizations for video streaming. It is developed to figure on any device and any platform provided, the device incorporates a browser. This may even be hosted on a smaller server while not carrying the video load. Splay acts as a far better resolution to the content management downside of establishments that need their own video streaming application. **[1]** The technology stack used parades new opportunities for contemporary development and continually expands the prospect of more measurability and upgrades .Here area unit the Dis-Advantages of Splay, As so much as mentioned within the paper, there are not any disadvantages however we tend to feel that the rear finish is bit a lot of optimized express.js and that we will improve the interface for far better expertise of the user by creating it mobile friendly, higher user rights management, self-updates etc.

So the Future scope of our paper is about the boundaries between PWAs and Web Apps are very light. **[1]** This may be explained with the dearth of definitions and with an alternative distribution of some technological aspects. To boot, some approaches don't serve multi-platform development because the primary goal however still contributes to that. An intensive study of this development is out of scope. Two approaches are unit noteworthy, though the conception of offline-enabling internet sites on mobile phones and also the hosted net apps by Microsoft, the conception fits mediate net apps and native apps, just like Progressive web Apps. **[3] [5]**

3. Technologies Used

3.1 Node.JS

Node. js is primarily used for not blocking, serves that are event-driven, and has single-threading nature. It's used for ancient websites and back-end API services, however was designed with time period, push-based architectures in mind. And we use Node.js in our project KLEF_PLAY as a result of Node.js is ab open supply server atmosphere and Node.js is liberal to use and it supports on numerous platforms (Windows, Linux, Unix, Mac OS X, etc.),it uses javascript on the server and that we will generate dynamic page content for our project by Node.js the most use of Node.js is it will collect knowledge|the info|the information and might add data, delete information and might modify information within the information Future seems to be sensible for Node JS within the front-end world because it feels like no side improvement is feasible while not Node. **[4]** Therefore it's best to use for our project, completely different Areas: Node JS internet development isn't utilised in numerous territories like embedded, computer science (AI) and Machine Learning (ML).

3.2 MongoDB:

MongoDB is the most commonly used database in the development industry as a Document database, MongoDB is a document-oriented NoSQL database used for high volume data storage. So it is best to use in our project to store a huge amount of data. MongoDB uses collections and documents in the place of rows and tables. MangoDb is faster than traditional databases like RDBMS, which is slower in comparison with the NoSQL databases. MongoDB supports deep query-ability, we can perform dynamic queries on documents using the document-based query language that's nearly as powerful as SQL. **[1]**

3.2.1 Why MangoDB?

-->Rich Object Model: This model is incredibly "object-oriented" and may simply represent any object structure in our domain. We are able to additionally index the property of any object at any level of the hierarchy — this can be brightly powerful!

->Secondary Indexes: Secondary indexes square measure a excellent construct in MongoDB. This makes it simple to index the associative property of an object kept in MongoDB albeit it's nested. This makes it very easy to question from the info supported these secondary indexes. **[1]**

->Good Replication and availability: MongoDB has a single type master model. This suggests we've a master node and a variety of slave nodes. Just in case the master goes down, one in all the slaves is not appointed as master.

->Native Aggregation: MongoDB contains an integral Aggregation framework to run associate ETL(Extract, rework and load) pipeline to remodel the info kept within the info**[5]**

. ->Schema-less Models:Each document in MongoDB will have a distinct structure and it's up to our application to interpret the info. Whereas this can be not relevant to most applications, in some cases the additional flexibility is very important.

Schema-less models mean that documents within the same assortment don't have an equivalent set of fields or structure, and customary fields in a very collection's documents might hold differing kinds of knowledge. MongoDB uses internal memory for storing operating sets leading to quicker time intervals. **[1]**

3.3 PWA:

A progressive internet application (PWA) may be a kind of application software system delivered through the online, engineered mistreatment common internet technologies together with HTML, CSS and JavaScript. -A Progressive internet App (PWA) may be a combination of internet and native applications providing such options as push notifications, offline operating, putting in on the homescreen, etc. Simply put, it's a technology that adds mobile app practicality to websites. **[3]**

A PWA may be an internet application which will be "installed" on your system. It works offline after you haven't got an online affiliation, leverage information cached throughout your last interactions with the app. If you launch the apps, they'll run in Chrome instead of having their own icon. **[3]** PWAs square measure designed to eliminate essential problems from slow networks to an entire lack of property. Such internet sites use the most recent web technologies to produce a quick, reliable, and fascinating user expertise.

3.3.1 The progress for developing PWA:-

-->Creating App shell. The applying shell plays a vital role in presenting the Progressive options of the app.

-->Implement a Service employee.

-->Adding Push Notifications.

-->Web App Manifest.

-->Analyse the applying performance.

We use PWA in our project as a result of the "PWA is that way forward for the online."

3.4 Video Encoding:

Video cryptography is the method of changing digital video files from one customary digital video format into another. The aim of this is often for compatibility and potency with a desired set of applications and hardware like for DVD/Blu-ray, mobile, video streaming or general video piece of writing. **[2]** The cryptography method transforms the video and audio information within the file and so will compression consistent with the specifications of the cryptography customary chosen.

4. Methodology

The methodology we proposed in the project is to make a Progressive Web App, shortly called as PWA, which works wherever we have a browser and has minimum system configuration. Also, we are making a video streaming web app which is a hybrid app that takes a major video streaming site like youtube and uses it for storage purposes. **[3]**

Video streaming works by breaking a video into tiny chunks and causation them via the online to induce reassembly and view at their final destination. The codec square measure visiting be a machine that compresses the covering into a bundle masses of|to avoid wasting} many houses. The instrumentality format may even be a freight car wherever these bundles square measure packed in. **[2]**

• Storage space: no additional downloads to a laptop computer, desktop, or phone.

• Access: anytime anywhere with consistent net association.

• sensible ol' buttons: pause, rewind, quick forward same approach as on a VCR.

• In control: away with spoon-feeding! Binge-watch your favorite shows.

• Variety: high-quality skilled on-demand videos, user-generated content (UGC), or live streaming events to cater to any or all tastes.

• Virality: video content is that the foremost shareable and attention-grabbing (hear, hear promoting folk!).

Live streaming is catching on in business, reworking the ways that social media promoting campaigns square measure run. Businesses square measure broadcasting live video streams for product demos and tutorials. Nonprofits square measure exploitation live streaming to bring major events to a good larger audience. Media organizations square measure streaming live news. Sporting leagues square measure exploitation live streaming to bring matches to their fans, where they are. Governments square measure exploitation live broadcasting to befits open-meeting laws. non secular teams square measure exploitation streaming media to grow their congregations.**[2]**

So a streaming platform development includes this practicality.

• Home page: The place to tell your users of the advantages they're planning to obtain, show them the capabilities, mesmerize them by the appearance and guarantees of amusement. The house section highlights movies, TV shows, sports programs, and alternative classes of content, and is made-to-order in line with every user's preferences (thanks to a sophisticated recommendation algorithm).

• Onboarding: unneeded to mention what proportion of onboarding advantages a product uses. however else would they grasp you provide the foremost effective functionality? however there's more! scale back your users' learning curve and guarantee they grasp each nook and cranny of your website. this would possibly create their keep persistent.

• Registration/Login: an important interactive component for a VOD service. offer a variety of however for your user to register/log-in:

o Facebook login

o Google login

o Mobile range for OTP verification

• User profile: It's sensible to possess somewhere to feel reception in. Once a home page, a user profile is just about this dashboard to navigate through the full journey, to not mention a wonderful place for you to urge insights (analytics-wise).

• Settings: lease users management the amount of their privacy is additionally a good plan of action. enable users to toggle between privacy choices for the audience and social platforms to broadcast to. **[4]**

Video shoppers would appreciate it if you build a streaming platform with these features:

• Search a video: Once your platform has fully grown, it'd be progressively laborious to travel searching the content. you'd form a strict system of classes Associate in Nursing an extended system of indicators for users to filter through. Keep filters for looking by location, broadcast topic, date, length of the video, genres, language, and most-watched, etc. Auto-correct and autocomplete can quicken the procedure, particularly for a mobile user. select one in all these tools to implement search on your web site.

• Video library: produce 'shelves' for viewers to treasure their best video experiences – 'viewed' history, to seem at later, favorites, likes, shares and such. The 'expiring' class indicates the episodes which will be prioritized on your watch-list.

• Video player & play experiences: Video players square measure your initial purpose of contact in conjunction with your audience, thus delivering the most effective doable expertise is important. **[2]** Viewers will produce a documented look and feel by customizing the subsequent elements: player emblem, watermark, a fine quality selector (crucial for a mobile viewer on a restricted information plan), information screen, share & enter choices, connected videos, list configuration, transfer button. Mind the integrations: analytics, proof.

• Subscription & interaction: Users ought to be able to obtain channels of their alternative. enable them to move with the content and community throughout a variety of ways:

o subscribe channels

o Like/dislike videos

o subsume videos

o Review and rate

o Share links to free video on websites

o Share the screen

o Live chat

• Payments: on-the-scene payments square measure indispensable for user retention. Users pay by payment sites or by their debit/credit card. the 2 payment models are:

o Pay to induce access to gated content (pay-per-view)

o Pay monthly subscription from your page

o Video creators ought to be treated lavishly to the foremost effective tools to create their keep pleasant and appreciated. they're going to require access to a specific studio and user analytics. **[2]**

• Video transfer and ingestion: facilitate transfer video from their multiple devices onto your website with video transfer. Video method|body process|bodily function|activity} permits the receiver of the video to want it in and process it for more use. Use the studio in Associate in Nursing passing terribly fairly ways:

o transfer files from the desktop and mobile devices

o Capture videos from the digital camera

o produce screen recordings in time period

o Launch live net events

o produce video displays with synchronised video and slides

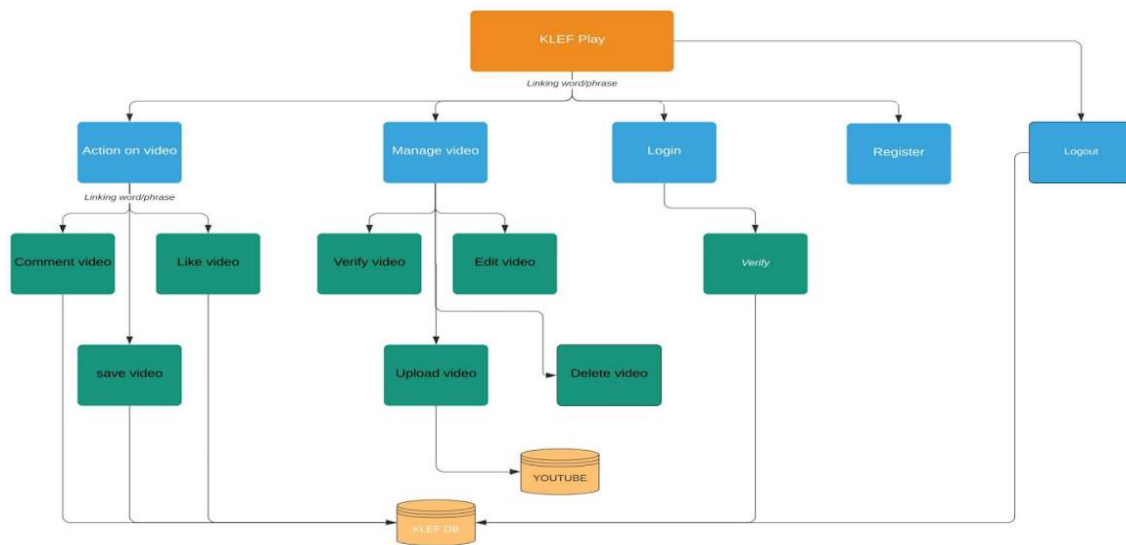o Import directly from YouTube at the press of a button

o Edit the recording

• Video management: Managing a media library, even Associate in Nursing outsized one, ought to be straightforward and straightforward. Content creators can like tools to rearrange videos and information, simply search media, edit the knowledge, moderate, produce playlists, etc.

• Analytics: offer the creators with a comprehensive pursuit system to live the engagement

We have worked on Client side, Server side and few other packages and read me files.

 On the client side, we have worked on different node modules that we require on the client side. Node modules are the multiple modules that we require from Node js on the Client side. All these node modules help us in proper and accurate working for the app. We have the source folder wherein we have all the Javascript and JSON files that we require in the working of the app. This also contains different components in it like the code for Dashboard, Form, Navigation bar, Signing In and Signing out, Uploading of videos and also how the video player should look. This makes us have a proper web app that has a great looking user interface and user design, which inturn helps in providing awesome user experience. We has also given the bootstrap CSS file that is required for the App and has index and Service Worker JavaScript files that will be used when required. And Lastly, in the Public folder, we have given the Index html page that opens wherever we have a browser and has also provided the manifest and logos for the App. **[3] [4]**
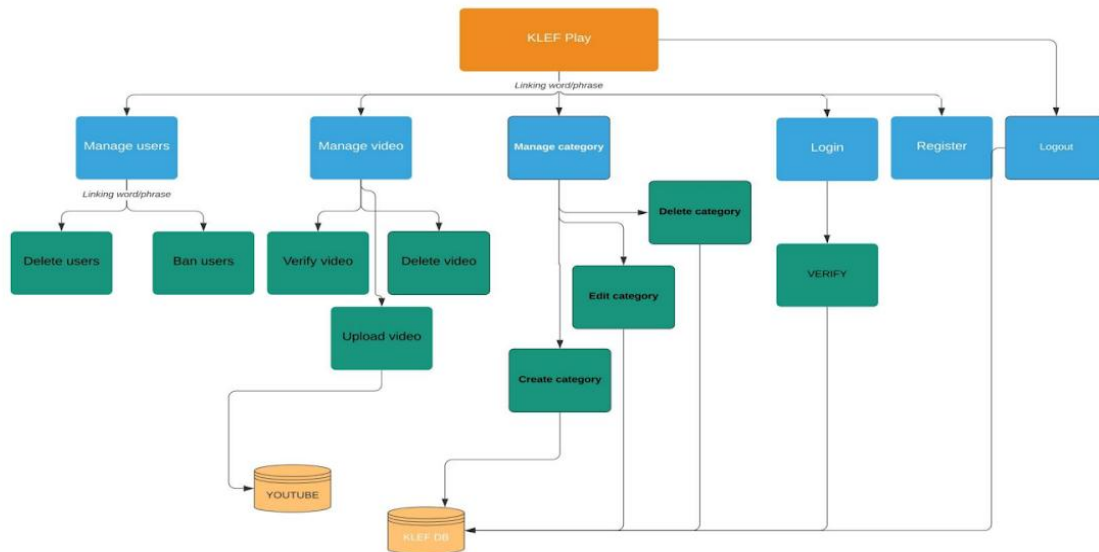
Fig-1: Client or User Use Case Diagram.

Fig-1 shows us how our Client or User side works. Based on the Use Case Diagram provided, through the Client side, KLEF Play manages videos, Actions on the video, Login, Logout and Registered Users. It Verifies Video, Uploads, Edits and Deletes the video in the managing the video section, that will be stored in the Youtube's database. Coming to the Actions on Video, we have the actions to Like the video, Save the video and Comment the Video and we have the Verification for the Login option. And all these categories have their data stored in the KLEF Play Database. This is how we depict the working of our Client side through the FlowChart or Use Case Diagram of the Client side given above. **[1]**

Now, In the Server side part we have worked on fervid things that helped us in having the best server side for a video streaming App. On the Server side, we have given the minimum configuration of the device for the Web App to work without any glitches. In the Helpers File, we have worked on giving the different options for the user to put a thumbnail for the videos and these options consist of taking a pic from the video itself or making their own thumbnail for their videos. We have a .gitkeep file that helps in uploading all the different media files. We used a middleware where we check the authentication for taking the embed files from the other sites and only with the help of that authentication, we will get the access for the video. For having the User information and Video Detail Schema, we have models folder that help us with all the required details. **[5]** We now have the Node Modules that we need in the Server which holds all the modules of Node js needed in Server or Admin Side. We have the package and package lock JSON files and also the App and Server JavaScript files here. Lastly, we have here the Routes folder that handles all the routes of the web app KLEF Play. This consists of SignIn route, SignUp route, Uploading route and the video listing route. This is what the server side comprises that helps in better working of KLEF Play. **[4]**

Fig-2: Server or Admin Use Case Diagram.

Fig-2 depicts the Use Case Diagram or Flow Chart of Admin Side. KLEF Play manages the Users, Videos and the different categories of the videos. Also, it maintains the Registering, Login and Logout options. On the Users side, it works on Deleting or Banning the Users and in the Videos part, it helps in Verifying the Video, Uploading it and deleting it and the whole Videos data is stored in Youtube's database. It also works on Creating, Editing and Deleting the multiple Video categories we have and helps in verifying the Login Credentials of the user. All of these Video Categories, Login, Logout and Registered Users data is stored in our KLEF Database. This is how our Admin Side will be working and we show all of its working in this Admin Use Case Diagram. **[1]**

We have provided some User Stories for our Web app KLEF Play. User stories play a major role as it is a casual, general explanation of a software feature written from the angle of the top user or customer. The aim of a user story is to articulate how a bit of labor will deliver a selected value back to the customer.

1) As a New user, I want to create an account, So that I can login to the account.

2) As a registered user, I want to change my password, So that I can keep my account secure.

3) As a website visitor, I want to subscribe to the channel, So that I get the latest updates.

4) As a mobile app user, I want to download the videos, So that I can watch offline.

5) As a registered user, I want to share the website link, So that others can register.

6) As a website visitor, I will save the wanted information, So that I can see when I need it.

7) As a mobile app user, I will give the review for the app, So that there will be a good rating for the app.

8) As a registered user, I will pay a monthly subscription from your page, So that I will get access.

9) As a mobile app user, I will use subtitles, So that I will have clarity about the voice in the video.

10) As a website user, I will add favourites, So that I get easy access to frequently used items.

5. Discussion on Results

KLEF play performs as a better solution to the content management problem of institutions desiring their own video streaming application. A complete package of ease to developers as well as users, it can be scaled and expanded in any way. Making it a Progressive Web App is just cherry on the cake. The power of PWAs can be seen in examples of the following implementations. **[1]**

- Treebo launched a PWA and saw 4x increase in conversion rate over a year. Conversion rates for repeat users saw a 3x increase. **[6]**
- Tinder cut load from 11.9 seconds to 4.69 seconds with their new PWA. The tinder PWA is almost 90% smaller than the native app. **[6]**
- BookMyShow's PWA takes less than 3 seconds to load and has increased conversion rates over 80%. The PWA is 54x smaller than android and 180x smaller than the iOS app. **[6]**
- Flipkart's PWA is driving 50% of its new customer acquisition. Nearly 60% of the visitors to PWA have uninstalled the native app to save the space. **[6]**

6. Conclusion & Recommendations

KLEF-play becomes a robust and trustworthy solution for small scale as well as large scale institutions and organizations for video streaming. Picking up good points from native as well as hybrid applications, KLEF-play stands between the two and provides better accessibility and more engaging experiences. **[1]** Not only, client's device space KLEF-play also saves more than 90% of server storage as well as complex calculations by using YouTube for streaming videos. If the organization plans to upload their own content or provide peer to peer video streaming facilities, the provided mathematical model will be used. The technology stack used opens new opportunities for modern development and always widens the perspective for further scalability and updates.

References

[1] Nilesh S Inkane, Siddhi A Kotak and Amitkumar S. Manekar "Splay: A Lightweight Video Streaming Application", 2019 International Conference on Innovative Trends and Advances in Engineering and Technology (ICITAET)

[2] John G. Apostolopoulos, Wai-tian Tan and Susie J. Wee "Video Streaming: Concepts, Algorithms, and Systems", HP invent[Online]. Available: https://people.cs.clemson.edu/~johnmc/courses/cpsc875/HPL-2002-260.pdf

[3] Andreas Biørn-Hansen, Tim A. Majchrzak and Tor-Morten Grønli1 "Progressive Web Apps: The Possible Web-native Unifier for Mobile Development", scitepress.org.

[4] Tilkov, S., & Vinoski, S. (2010). Node.js: Using JavaScript to Build High-Performance Network Programs. IEEE Internet Computing, 14(6), 80–83.

[5] Malavolta, "Beyond Native Apps: Web technologies to the rescue!", 2016 ACM.

[6] PWA stats, Accessed on Oct 15, 2019 [Online]. Available : https://pwastats.com