Advances in Computational Sciences and Technology
ISSN 0973-6107 Volume 12, Number 1 (2019) pp. 23-36

http://www.ripublication.com

Hybrid Web Application using Content Management System App used by all platforms

Navyashree S
MTech Student, ISE,
R.V College of Engineering Bangalore India.

Rashmi R
Assistant Professor, ISE,
R.V College of Engineering Bangalore India.

Abstract

Developing a content management system one of the hybrid app that delivers the required data to the subscribed users at specified time. The subscribed users with different platforms like Android, Windows, IOS, Mac OS, AmigaOS, Blackberry can receive the data simultaneously to all subscribes at specific time. This is useful for the developers as the single app designed is useful for all platforms and it is also cost effective.

Keywords: Content Management System, Hybrid Platform, Restful Services

I. INTRODUCTION

Since the previous few centuries permeation of mobile devices is increased rapidly. Also progress of customers who use mobile devices has directed to evolution of numerous types and varieties of devices which might have different platforms. Due to these versatilities the demand for developers who are expertise in different programming languages are rapidly increased all over. Hence the mobile app development has various challenges, selecting the right development tactics is very much essential and a big challenge. Selecting the relevant techniques which is suitable for all the platforms, and also saves development time, resource and money is very much essential.

When we are building the mobile application one that strikes in mind may be 1: Native App, 2: Hybrid App. Based on the customer requirement we have to design the application. The definition of native application is, "an application that is developed for the use on a particular mobile device or application development platform." Native apps are written for a specific device operating platform to fully exploit all the features and coding of the platform's language. Usually, most native Android applications are

written in either Java or Kotlin using Android SDK. Android uses Android Studio for native application development along with these two Software Development Kits. The definition of hybrid application is, "an application that uses common code to utilize all the common features of various mobile device operating system platforms". Hybrid apps are written for a pan-device platform to utilize common features of app development platforms. Hybrid applications are written in a combination or standalone ref languages of HTML, JavaScript, and CSS. You can use cross platform tools like Phone Gap, Titanium, Cordova, and Ionic to build hybrid applications which will run in web view. The Hybrid app provides a way to develop the multi-platform application using single codebase. Choosing among this two is very much required.

Native application development is the easiest choice for most app developers. As it is easy for them to develop and they can fully utilize all the platform features. Ideally, when you want to take full advantage of hardware, performance, User Experience and in-built features for a large user base to update quickly and frequently, you use native app development. Pros and cons of native application includes

Figure 1: Native app

Pros:

- Developer augments code with native SDK
- Rich User experience – full possible of the platform can be leveraged which will provide great user experience
- Higher Security
- Have native API access easily like Sensors, Camera, Push Notification

Cons:

- If you are targeting to develop app in multiple platform, you need to rewrite native code for multiple platforms
- Take times to develop application compared to Hybrid app

- If app is developed in multiple platform with specific platform, Development cost will be higher
- Require specific knowledge for each platform

Hybrid application are now on the rise, thanks to simplifying their coding languages and easy integration with devices and web view. The hybrid application development when you have budget constraints, do not want a lot of features in your app or you have large content driven app. Pros and Cons of Hybrid application includes

Figure 2: Hybrid app

Pros:

- Can be easily deployed across multiple platform (Android, iOS, Windows)
- Hybrid app developer can develop same application using single codebase for multiple platform, it will have lesser cost compared to Native app development
- If you are targeting to capture market as earliest, you should choose hybrid app development
- If you developed app in Android and iOS only and now you want to developed in Windows platform, it can be easily scalable for newer platform

Cons:

- Performance is the biggest disadvantage while developing apps in hybrid because application renders in WebView
- It has limited access to hardware resources like sensors, customizing camera. For customization plugins, hybrid developer needs to rely on native developer for each platform.
- After fixing a bug, required to test feature in all platforms as it may impact on another platform.
- Sometime building Hybrid app costs equivalent to Native apps when you are trying to build components like native app.
- UX of Android and iOS will be same, so loyal users might not get pleased. It's hard to customize the hybrid app based on the platform and it will effect on cost.

Content Management System (CMS): It is defined as managing the data that is stored inside the server or any storage device as separate objects. It is very much essential to manage the data and send them to the subscribed user. Also it is required to maintain the multicast networking such that same data can be sent simultaneously to multiple users who have subscribed to the application.

Content management is defined as set of process and techniques that is majorly responsible for content management which includes collection of data, handling the data and producing the content in any form or type Content can be of any form like which may be structured or unstructured text (document or pdf), multimedia format (audio, video) or other file format which undergoes content life span and requires management. CMS is a collective process [6] which usually contains the below rudimentary characters and duties:

- Creator: major responsibility includes generating and managing the document where he can edit.
- Editor: changing the format or style in which the content is delivered would be the major responsibility.
- Publisher: the major responsibility is to publish or issue the content to the user.
- Administrator: majorly involved in providing the right access to various end-user, he will assign the file permission to user or to group. Only those users who are provided access right are allowed to modify the respective files.
- Consumer: the individual uses the content that is been published.

Figure 3: CMS general perspective

II. OBJECTIVES

- Developing content management system using hybrid platforms for sending similar data at a specific time for enrolled users.
- A CMS for managing content in mobile app(s), possibly for several devices (iPhone, iPad, Desktop, Laptop) and/or several platforms (iOS, Android, Windows, Mac OS) – from a single developed app.
- Managing the contents of the application without coding a general person can manage editing and storing the contents.
- Hybrid platform application enables more users since app can run on any Operating System.
- Designing cost efficient and user friendly application with faster performance.
- To enable applications to run on multiple platforms.
- To enable interoperability, portability, usability.

III. LITERATURE SURVEY

1. BUILDING A WEB CONTENT MANAGEMENT SYSTEM(CMS)

The author has designed a CMS for his university to manage the static HTML pages of their web site. [1] It investigates different present open source Web CMS and its characteristics. Web CMS major purpose is to permit novice users who do not have programming expertise can simply change and edit the website's content without any technical expertise. WCMS products has integrated many different complex functionalities that enhances the user interface. All WCMS need some training on installation and how to start with the application. The research explored in the paper is to implement a WCMS from already existing choices which is made in comparison from building it from scratch. This paper exposes building an entirely a new WCMS, including the advantages and disadvantages

The author here discloses how to build a CMS which is simple and can be used for managing the content rapidly in the existing website which is been developed by HTML. The example contains the newscast segment of the CS dept. which requires regular updation and since application uses HTML page, a person with technical knowledge of HTML and CSS programming language is essential to edit the content which might include the small portion also. The main goal of paper is explain integration of CMS engine which might can be combined in the News and Events section page of the University's CS dept, such that procedure of managing and editing the content would be much fast and easy for the dept. The central drive of this project is to get manageable and easily editable content managing interface which contains commonly used CMS functionality which is appropriate for minor and simple websites, such that any user who is not from technical background can manage and maintain the website content. A novice user can simply for adding, editing and arranging the content

on the website by the amusing text editor combined with the CMS simple engine.

2. CONTENT MANAGEMENT SYSTEM FOR ONLINE-GOVERNMENT PORTAL

This paper [2] explains the event based architecture and its various benefits and compared with the platform of WordPress, by using technical enhanced answer for building Online-Government gateways. It gives some hard, segmental and dynamic basis, which is adapted to replying for different new encounters and good methods and techniques which includes management of information within the Website, by not conceding the enhancement and development of the website for the system which is appropriate for minor teams of the project, which is the elucidation aimed at fast development website, which can reduce the prices and time required for the development activities. The result is been employed in the platform that currently provisions for the portals of web.

3. JooMDD: A Model-Driven Development which is suitable Environment for Web Content Management System Extensions

Designing the software additions using Web Content Management Systems (WCMSs) [3] which can include Joomla, WordPress, or Drupal can be a hard and tedious procedure which requires more time. The paper it presents JooMDD, and situation using model-driven-development of the software allowances for the WCMS Joomla. JooMDD permits quick progress of standardized software additions which requires condensed technological familiarity of Joomla. From this it indicates that un-experienced developers can create their own individual practical WCMS extensions. This shows that a model-driven method which would be more appropriate in the domain of WCMSs.

Practical extensibility is one of the main reason of using the WCMS platform for dynamic websites compared to other several standard extension types. Although we can use other extensible APIs, one can apply the extension instead of altering the platform. Additionally, we can use the API functionality, that is already developed and extension code can be written and condensed, while in other way it permits further quick and rapid development. Knowledgeable extension developer uses these mechanisms to surge the rapidity of the development stages. Non Knowledgeable developer exposes fewer usage using that API, since most of them are not familiar with the functions which are accessible and when and how to adapt them. Owing such type of problem and also point that many WCMS extensions, are usually free from their original platform, which may hold more duplicate code, the author suggest a model-driven-approach for the progress of WCMS extensions quicker and further simply the context to programming language that is manual. One can accept the formation of design-models and further succeeding code modifications rise development rapidity individualistically from developer's knowledge.

4. Intelligent Content Management System used For Discussion Forum

Drawbacks in recent mass media communication connected outfits are conversed in the paper [4] Different model for the use of online forum has been anticipated for the purpose of eradication of these drawbacks. Algorithms that can be used for the typical model is been suggested. Approaches for execution of the algorithms will be shown here. And also the software architecture that is been showed for implementing the work is been showed here.

The key goal of this forum content management system (CMS) development is that one can achieve and accomplish laidback incorporation of the present competences with the help of forum engines. Hence, building this software using the ethics of micro-service architectural style is a key advantage. This Software probably fetches a single one point service that covers majorly required data and logic, interaction of the both takes place using the most suitable API called REST-FUL API the most commonly used API for micro-services. The architectural style allows to separate software system, parallel to the leaving the opportunity of rapid and easy incorporation with other existing systems.

5. Web content management system (CMS) use of architecture, concepts and application

The first and the most important goal of this study is to propose design of a new content management system. Important fields such as architecture, database design, various concepts and user interface design are discussed. Proposed concepts and features are demonstrated on author's genuine solution, a content management system called Urchin CMS. This project has already started in January 2011 and is still being continuously developed and improved.

The system serves as both an experimental platform and a valuable tool for website development. The limiting factor for development is time, as author both has a full-time job and studies at the university.

Secondary goal of this study is to discuss multiple ideas for key concepts. This contains comparison of the most important concepts with other approaches and existing solutions, e.g. massively used open-source content management systems, such as Drupal or Joomla. Pros and cons of these systems are mentioned as one of the reasons leading to the development of Urchin CMS. This study could also serve as a rough guide for architects or developers of content management systems. This work covers all important areas of the design and architecture.

IV. METHODOLOGY

In our project we are trying to implement an application that sends the shlokha to the subscribed users periodically around morning and evening for their mobile or laptop apart from different Operating system. We have to send the content that is shlokha daily to the users. Here we can store different shlokha as different objects inside the server

or any other storage devices and manage all the subscribed users and send those data to those subscribed users at a particular interval so we make use of CMS, as the content has to deliver for different subscribed users who may use different platforms so we have to make use of hybrid app development.

The administrator who doesn't even knows coding can manage the data in our application so CMS plays a very important role for doing this. The admin can view modify edit and delete the data content using just click option it becomes a very good user friendly application. The admin can also manage the users at what time the data has to be sent and which day what content has to send. We have to map each object to a particular day and also maintain a scheduler for sending the contents to end users.

As we are using a hybrid platform to design this application there is no hard for development because one app may serve to different platforms so no need of developing different User Interface (UI) for different OS as the UI May vary from one operating system to another. Hence coding time is reduced. Hybrid-Platform is one of the finest methodology that can be used by consumers and developers. Let us take look at the following requirement necessary:

- The essential of using hybrid-applications
- Capability that hybrid-platforms can be used to assist app development and its benefits and drawbacks
- Most Popular hybrid platforms

The only Need of hybrid application development in the earlier days of application development, there were only limited quantity of devices that were running on particular platforms with particular programming languages.

However nowadays developing the application that fits to a specific device requires extra effort, cost and development time, since the developers has to account to know various technologies that are specific to each mobile device like android, IOS, Blackberry, Windows. In the other way round hybrid app development can be termed as "Fit for all with single code base" which requires less development cost and time. Types of mobile application includes.

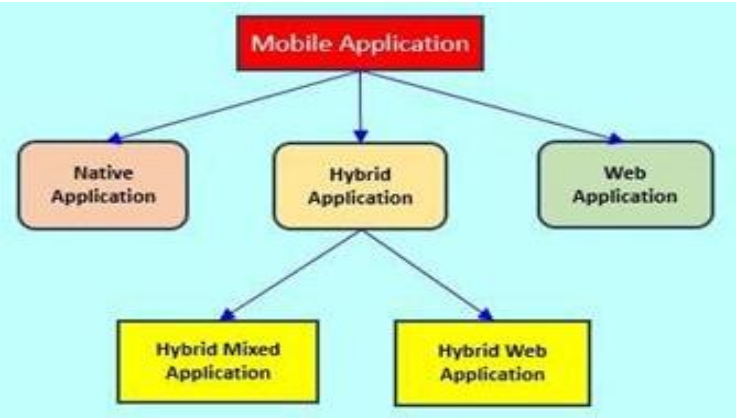

Figure 4: Types of mobile application

Native-Application-Development (NAD)

Native apps are basically used to run in a specific device like android requires separate programming language whereas IOS requires different language. The application consume entire device access also we can create complex User Interface by using native- framework backing a particular programming language by platform provider for example. Android requires java and Android SDK technology, iOS requires Swift, objective C and X-Code technology, Windows platform requires C# and Visual-studio IDE. Apps are shared and distributed using app store and play store such that it would be easier to reach the customers. From commercial viewpoint, developers want to wage great costs when developing native-application. Building native-applications needs skill with the specific SDKs, languages on particular platform and also essential in managing code base for individual platforms. The major drawback in native application development includes adapting new technologies. Working individually on target platform for specific platform code base reduces down the output process. Examples of native application includes games and augmented reality.

Web-Application-Development (WAD)

Web-applications majorly developed for running on mobile device browser. Building this mobile web application developers needs to be specialized with specific skills which includes HTML5, CSS3 and JavaScript to. Web applications development requires single code base which decreases up-gradation procedures. These types of applications are usually not shared using native stores and not installed as packages on respective devices. The web applications is not decent optimal way for the complex User-Interface Examples includes Business app, enterprise app, financial applications.

Hybrid-Application-Development

To overcome the restrictions of native application or web apps, this hybrid application development is been introduced. It's not much difficult in case of Hybrid application development to enhance software development process across platforms one can also ensure to minimize price of tools, repairs and resources. Hybrid application development can be divided into two categories Hybrid-mixed and Hybrid- web-application development.

Hybrid-web-applications are developed using HTML five or JavaScript or CSS three technologies and also enclosed within native-platform web container ie Web-View that is present in Android. Instead to run it on mobile browser, these hybrid web applications are downloaded and installed on device through market place similar to that of native application. Some of the best hybrid application development platforms includes Cordova, Sencha & Kony.

Hybrid–mixed-applications is usually developed using technologies like C#, JavaScript. Hybrid mixed apps are provided through market place. To build native application that might contain much complex and rich User Interface, high device

access and it can target on multiple platforms and nevertheless it also want everything at a defined one specific code base, hybrid-mixed-native-application is one of the finest approach to take for the development. One of the best hybrid mixed application example is Xamarin.

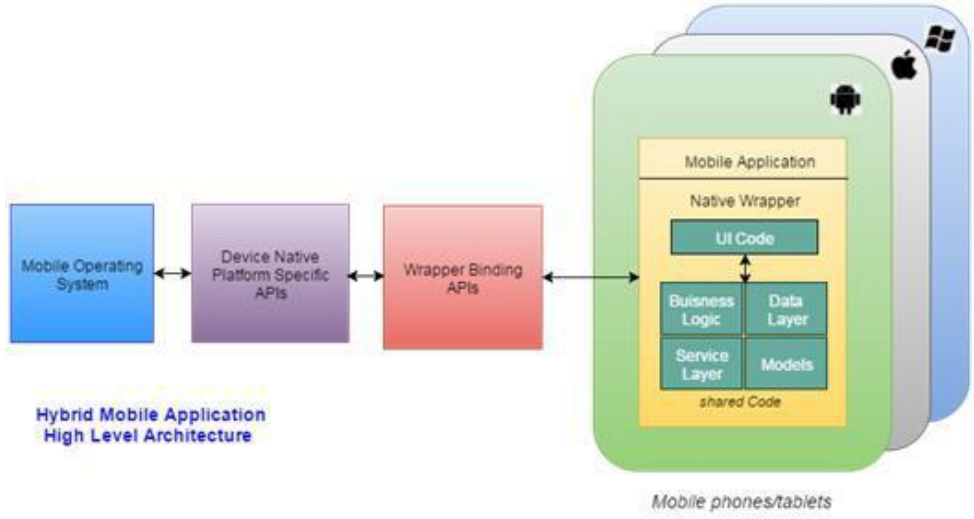

Figure 4: Architecture of the application

When to use which is little confusing the below table gives the idea of when to use which platform for developing the hybrid application.

Table 1: Comparison of different Tool

Platform	Mobile Development Platform	Programming Language	Integrated development Environment	Emulator Support	Tool Cost
Xamerin	Android, IOS, Windows	C#	Visual Studio/ Xamarin Studio	yes	Licenses available
PhoneGap /Corodova	Android, iPhone, Windows, Tizen ,Blackberry, Symbian, Plam, Bada	HTML, CSS, JavaScript	3rd Party Tools	3rd party tool	Free commercial license
Sencha	Android, IOS, Blackberry, Kindle, Bada	HTML, CSS, JavaScript	Hybrid via native shell for each platform	yes	Free commercial license, Paid OEM and embedded system license
QTSDK	Symbian, Android, IOS, MeeGo	C++, QML	Qt Creator	yes	Free commercial license
Unity	Android, IOS, Nokia Symbian	C#, JavaScript, Boo and other .NET based languages	Unity Editor/Visual Studio/MonoDevelop	Remote used to simulate remote interaction	Free commercial license

V. RESULTS

Following are the snapshots of working web application:

There is landing page where there are 4 categories of content that is picture of god. Upon click of a god's picture, the related list of singers who have sung song will appear. Upon clicking on the singers list of songs categorized by those singers will appear. Any user can view the content, also view the lyrics and download lyrics as well as song in MP3 format. Registered users upon clicking the subscribe button, subscribed users get notification of the lyrics of song in the form of email. All this content will be managed by admin; admin has the privilege to add the content to the CMS. Admin by selecting the category, the name of the singer, song and lyrics can upload the song. After successful addition of the content he gets the message successfully added. Admin can also view the content that he has added and also delete the content if it is not required.

Figure 5: Landing page of the application

The figure 5 shows the home page of the web application. This page is common for both admin and user.

V. CONCLUSION

Mobile applications has various types and platforms which is technology independent and requires separate sill sets, so while developing the application that needs global platform requires proper choosing of development tactics hence choosing of right technology is very essential Proper planning has to be made before developing the applications .Scalability and platform independency is the major factor that has to be

considered while developing the application. Write the code once and run in any mobile or web platform is the motto of this project. As we has seen the screen shots of the project this application can be used in different platforms. So it is very essential to design a platform independent application.

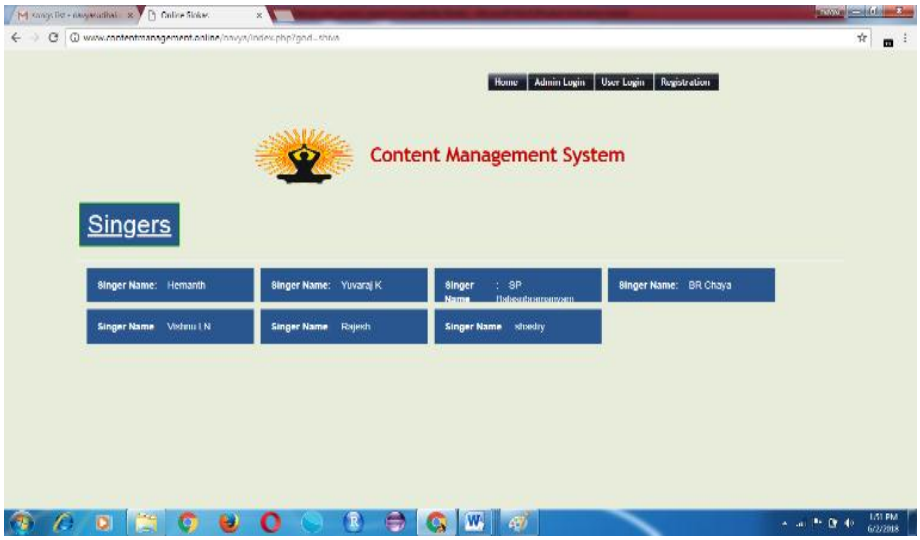

Figure 6: Singer's category page

The figure 6 shows the category of singers upon clicking on any one of the god's picture the respective singer's list can be viewed as shown above.

Figure 7: List of songs of singer's category.

The figure 7 shows the list of songs of a particular singer. Upon clicking on any one of the singer name the respective list of songs that belong to that particular singer will be shown.

The figure 8 shows the admin panel where in it shows all the content that have been uploaded and the respective delete option if the content is not required admin can delete the content.

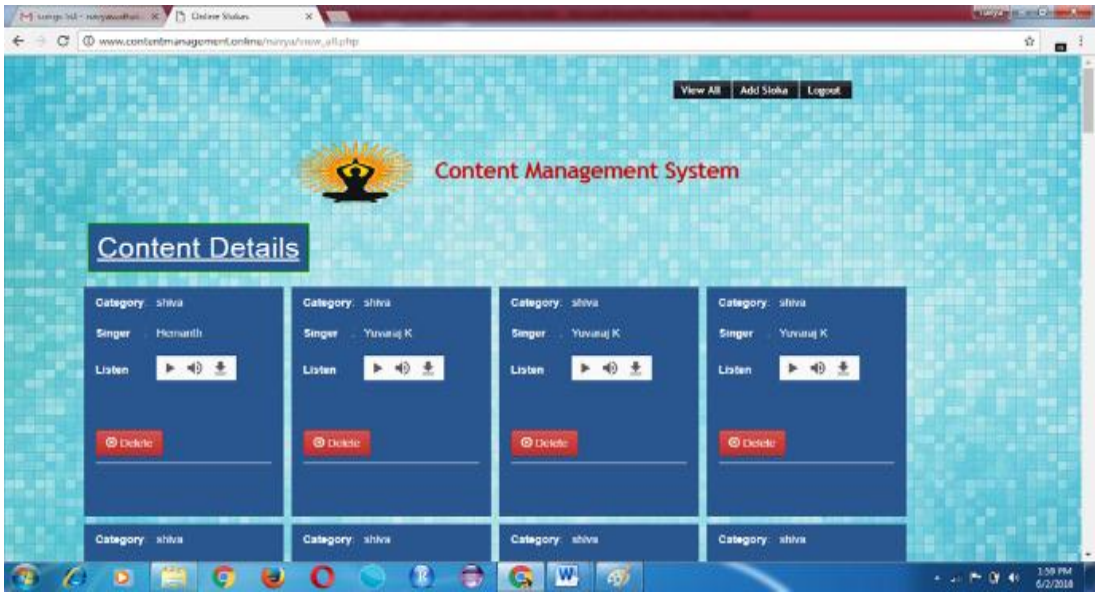

Figure 8: Admin Dashboard.

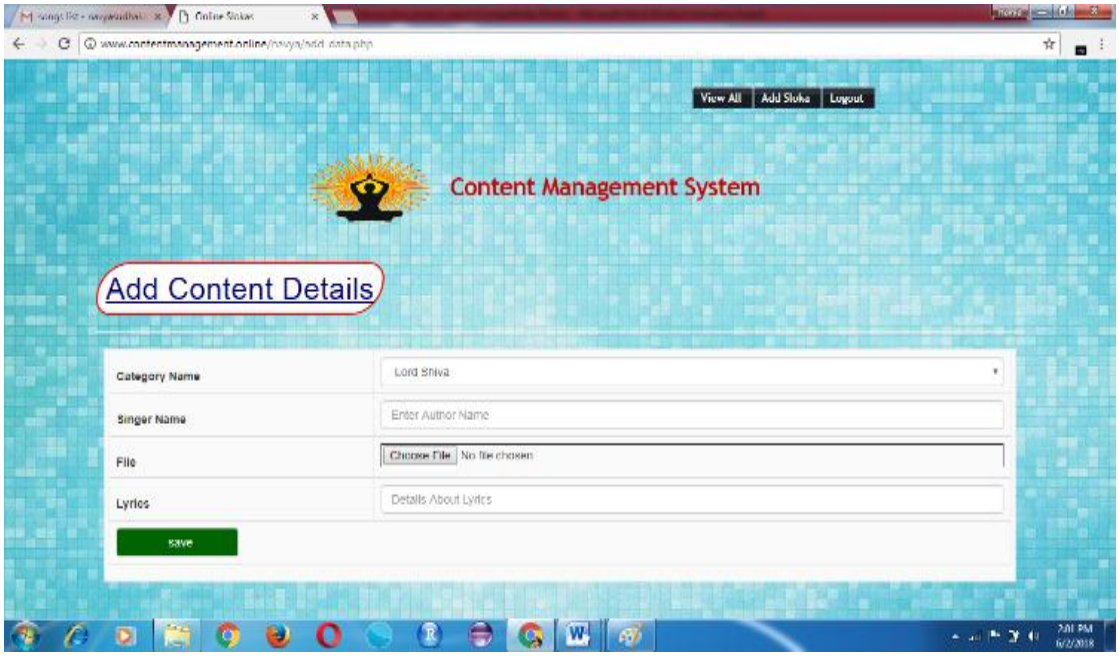

Figure 9: Admin content adding page.

REFERENCES

[1]. "BUILDING A WEB CONTENT MANAGEMENT SYSTEM" A Thesis Presented to the Faculty of San Diego State University by Rima Vyomeshbhai Shah.

[2]. "*Content Management System for e-Government Portals*" by Cláudio Esperança Escola Superior de Tecnologia e Gestão Instituto Politécnico de Leiria Leiria, Portugal Unidade de Ensino a Distância Instituto Politécnico de Leiria Leiria, Portugal and António Pereira Centro de Investigação de Informática e Comunicações, Escola Superior de Tecnologia e Gestão Instituto Politécnico de Leiria Leiria, Portugal INOV INESC INOVACAO Instituto de Novas Tecnologias Leiria, Portugal.

[3]. "JooMDD:A Model-Driven Developmen Environment for Web Content Management System Extensions" by Dennis Priefer KITE - Kompetenzzentrum für Informations technologie, Technische Hochschule Mittelhessen, Germany and Peter Kneisel KITE - Kompetenzzentrum für Information stechnologie, Technische Hochschule Mittelhessen, Germany.

[4]. "Intelligent Content Management System for Discussion Forum" by Kozko A. Institute of Informational Technology Chelyabinsk State University Chelyabinsk, Russia.

[5]. "Web content management system: architecture, concepts and application" a master thesis by Bc. Josef Kunhart.

[6]. https://www.xoriant.com/blog/mobility/mobile-application-development-hybrid-platform.html.

[7]. https://www.contentful.com/r/knowledgebase/mobile-cms/

[8]. https://www.3pillarglobal.com/insights/when-to-take-a-hybrid-approach-for-mobile-app-development

[9]. https://www.knowarth.com/develop-app-native-hybrid-platform/

[10]. CMS Architecture First Edition by Shailesh Kumar Shivakumar. © 2017 by the IEEE Computer Society, Inc. Published 2017 by John Wiley & Sons, Inc.