

Developing a Web-based Agro-Application¹

Lakesh K. Sharma, Rishabh Khanna, and Hardeep Singh²

This publication discusses the general concept of web and mobile application development and provides knowledge and stepwise instructions to develop an application in agricultural systems. The primary audience is researchers, Extension personnel, consultants, and students who are interested in developing and using web and mobile applications for decision-making.

Introduction

These days, an application, whether a web app or a mobile app, is an important medium of communication and analysis, as well as a base of business modeling, to transfer information for any domain. For example, professors post assignments on UF EDGE portals, and students submit their projects online. Such apps are categorized as Educational apps. Many business applications sell products or provide services like food and goods delivery. These applications come in handy when we want to track our delivery packages or compare product prices across stores. Applications can be categorized as (Yevtushenko 2016)

1. Gaming
2. Business
3. Education

4. Lifestyle
5. Entertainment
6. Utilities
7. Travel
8. Book
9. Health and Fitness
10. Food and Drink

In agriculture, with the use of web or mobile applications, farming has transformed into smart farming, like Farm Futures (<https://www.farmprogress.com/farm-futures>), an education application that provides information about various agro-based companies, products, and blogs. Smart Irrigation (Mbabazi et al. 2015), Crop Advisory Systems (Fraisie, Perez, and Andreis 2015), and AgroMobile (Prasad, Peddoju, and Gosh 2013) are examples of utility apps that help in crop production and maintenance among others. Such applications have been helping farmers and researchers in agriculture. An article by Lariviere (2011) elaborates on how farmers like Mr. Shinpuku of Japan have been using computer applications in farming for the past decade. These applications help Mr. Shinpuku in crop

1. This document is SL513, one of a series of the Department of Soil, Water, and Ecosystem Sciences, UF/IFAS Extension. Original publication date April 2024. Visit the EDIS website at <https://edis.ifas.ufl.edu> for the currently supported version of this publication. © 2024 UF/IFAS. This publication is licensed under CC BY-NC-ND 4.0.

2. Lakesh K. Sharma, assistant professor, Department of Soil, Water, and Ecosystem Sciences; Rishabh Khanna, former master's student, Department of Computer and Information Science and Engineering, UF Data Research Lab; and Hardeep Singh, assistant professor, Agronomy Department, West Florida Research and Education Center; UF/IFAS Extension, Gainesville, FL 32601.

The use of trade names in this publication is solely for the purpose of providing specific information. UF/IFAS does not guarantee or warranty the products named, and references to them in this publication do not signify our approval to the exclusion of other products of suitable composition.

The Institute of Food and Agricultural Sciences (IFAS) is an Equal Opportunity Institution authorized to provide research, educational information and other services only to individuals and institutions that function with non-discrimination with respect to race, creed, color, religion, age, disability, sex, sexual orientation, marital status, national origin, political opinions or affiliations. For more information on obtaining other UF/IFAS Extension publications, contact your county's UF/IFAS Extension office. U.S. Department of Agriculture, UF/IFAS Extension Service, University of Florida, IFAS, Florida A & M University Cooperative Extension Program, and Boards of County Commissioners Cooperating. Andra Johnson, dean for UF/IFAS Extension.

surveillance and detecting crop problems like infection. Research by Csótó (2015) showed that 45% of farmers in Hungary use a smartphone, 14% use tablets, and 72% use a desktop or laptop computer for agricultural purposes, just like Mr. Shinpuku. A survey conducted in the United States, reported by Nelson (2019), showed that 49% of farms use computers and around 80% have internet access. Today, if a farmer uses mobile devices, then utilizing different agricultural apps would make their farming business more efficient. In light of this information, it becomes essential for researchers, Extension personnel, consultants, students, and so forth to be aware of the development process of an application.

Pre-requisites for Application Development

Before developing an application, anticipating how the application will be used is critical. The preliminary questions to be answered should be—who will be using the application and how will it be used? These questions will help figure out the right platform—web, mobile, or both—supporting the application. Some applications are specific to mobile platforms and cannot be used on computers or laptops and vice-versa. Mobile-based applications are either iOS or Android applications that are available on different operating systems stores like Apple Store or Google Play Store. Some applications are only web-based but can be used via mobile browsers. Other factors, like budget and reach of the application, also play an essential role in deciding which platform to consider. A mobile application might be costlier than a web application, but a mobile application might serve more users than a web-based application. The aim of the application will incur different infrastructural (hardware and set-up systems) and resource costs. For a blogging application, few infrastructural resources are required; hence, such applications are cheap. If the application requires using a database or the application’s algorithm requires statistical modeling, then web developers and statisticians will incur an additional cost.

Compared to a mobile application, web applications have the advantage of giving “one solution” for all platforms, meaning a website can be used from any device having a web browser like Google Chrome, Safari, or Internet Explorer. These web applications do not require separate development for iOS, Android, or Windows. Therefore, the following section discusses the basic concepts and development of web applications.

Building applications can be categorized in two broad approaches:

- Building an Application from Scratch—This route of application development requires programming knowledge and coding for using various web and mobile development technologies.
- Building an Application using CMS—This route requires less or no programming knowledge and is usually preferable in cases when a lot of custom logic (i.e., creating an algorithm) is not required to build the application.

Table 1 shows a brief comparison between the two approaches.

Table 1. Comparison between building an application using CMS and building an application from scratch.

Using CMS	Building from Scratch
Does not require too much coding knowledge	Require knowing different technologies, their pros, their cons, and coding
Best option for quick building	Not very good when a quick build is required
Easy to maintain	Hard to maintain
Less flexible	More flexible
Using custom logic is complex	Building custom logic is no different than coding the rest of the application
Uses plugins which may be unreliable or insecure	Security can be self-managed without being dependent on third party plugins

Building an Application from Scratch

For building the app from scratch, various programming languages and technologies are available in the market, and each has its advantages and disadvantages. Based on the app use case, technology can be selected.

User interfaces for websites are usually created using HTML, JavaScript, and CSS. A detailed description of writing and designing a basic web page, developing visual elements for it, and understanding the difference between good and bad websites is provided in Ask IFAS publication #AEC566, “Writing and Designing a Basic Web Page.” This phase is called building the front-end of the application. The algorithm, or programming logic, is developed using programming languages like JavaScript, Java, C#, and PHP, among others. This is usually referred to as the back-end of the application. This part handles how the data is transferred from the user interface to the database and all the calculations that go inside the system.

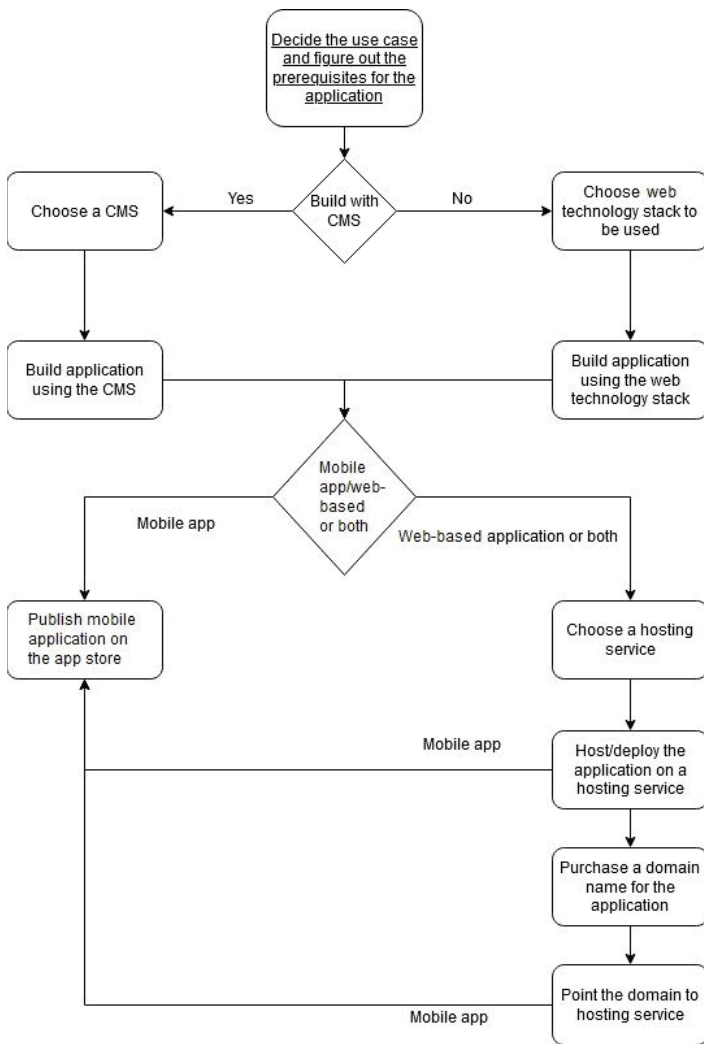


Figure 1. Application development flow.
Credits: Rishabh Khanna, former UF/IFAS

To explain the building of the app, it is helpful to discuss an example. Consider the problem statement: a web app is required to predict the N rate for potatoes based on certain conditions like region, fertilizer cost, crop rotation, and yield. Use case of such an application would be use by farmers for applying N-based fertilizers for the potato crop. As per the use case, a web application was built using the programming language, JavaScript, and web frameworks, React, Express, and Node.js. PostgreSQL was used as the database. Selecting these technologies was based on market trends and the difficulty level of developing the app. These technologies, together, are often called a technical stack of the app. If one wishes to learn these technologies, various online platforms like Udemy, Udacity, and Coursera have various tutorials and courses offered by experienced professionals designed for individuals with a non-programming background. An alternative to learning these technologies can be to hire a software developer.

Once the problem statement, use case, and technical stack have been thought of thoroughly, the next step is to define the application logic. Application logic is generally referred to as the algorithm which is underlying the application. For the N rate calculator, application logic is based on a paper by Sawyer and Nafziger (2005). The logic works by calculating the Maximum Return Nitrogen Rate (MRTN) based on the selected region, organic matter content, cover crop, crop price, and fertilizer cost (Sawyer 2019). These input parameters are used to construct data in the background and then find MRTN out of the newly engineered data.

This business logic (algorithm for calculating MRTN) is coded using the technical stack. The user interface is developed for web using React, HTML, and CSS (Figure 2). Another user interface is developed for Android and iOS using React Native. After development, the app is thoroughly tested. Testing involves running all the possible use cases, often termed as test cases, and checking the behavior of the app. For example, a use case can be when the user inputs the wrong text and the app provides the desired output.

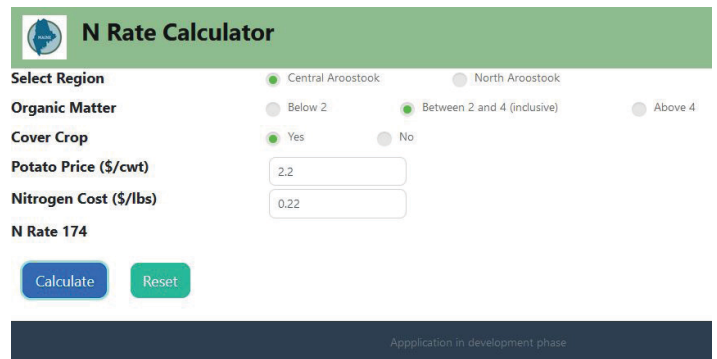


Figure 2. N Rate calculator for potato crop.
Credits: Lakesh K. Sharma and Rishabh Khanna, UF/IFAS (N Rate Calculator – <https://rishabhrrk.github.io/nrate/>)

After testing, the application can be made public by deploying it to the cloud. A cloud vendor is selected for deploying an application on the cloud based on the requirement and financial constraints. Cloud vendors include Microsoft Azure, Google Cloud Platform, Amazon Web Services, Heroku, DigitalOcean, and others. Deployment involves various steps, and these vary according to the vendor. The vendor often provides a tutorial for the deployment, and the same can be referenced. Often universities have their own cloud, and these can also be used to deploy the application. For example, the North Dakota State University Potassium Recommendation Calculator (Franzen 2018) is deployed on the university cloud and not on any private vendor.

Building an Application using CMS

The approach using Content Management System (CMS) is very generic. Using CMS is quite similar to creating a PowerPoint presentation. A CMS is ideal for building a basic application that does not involve high user interaction or data processing. A CMS is software that allows quick publishing and editing of the content on a website without the need to alter the site's code. Applications built using CMS do not involve coding or minimalistic coding; that depends on which CMS is being used. With such systems in place, an agricultural blog, classroom lecture video website, or even a small business's web portal can be created.

With the use case in mind, the next step would be choosing an appropriate CMS. There are many commercially available CMS, such as Joomla, Drupal, Zyro, WordPress, Jimdo, and Sitebuilder, all of which have similar offerings. The most popular is WordPress, with nearly 35% of websites on the internet built using it, according to WPBeginner (2024). WordPress is proven to be better than others as it gives an option of coding when required to incorporate customization and has been one of the oldest and evolving CMS in the market. The following steps should be followed for building web and mobile applications using WordPress (n.d.):

1. Signing Up for a WordPress account—Create a WordPress account. The account username and password together ensure that you are the developer and will take you to the developer space. If the username and password get in the hands of a malicious user, then they can modify your application. Hence, these should be kept very secure.
2. Selecting a Theme—CMS mainly focuses on building the interface rather than the working of the application. See this blog post by Rahul Sahu for locating agriculture-oriented themes on WordPress: <https://www.gomahamaya.com/farming-wordpress-themes/><https://www.gomahamaya.com/farming-wordpress-themes/>.
3. Changing the website—After selecting the theme, it is time to change the contents. WordPress provides the template with some default contents like pictures or text, which require customization depending upon the user's needs (Figure 3).
4. Adding a Web Page—New pages can be added or removed (Figure 4). Creating pages requires adding text and new media, like pictures, videos, and even audios, if necessary. This design phase of website development takes the longest time.

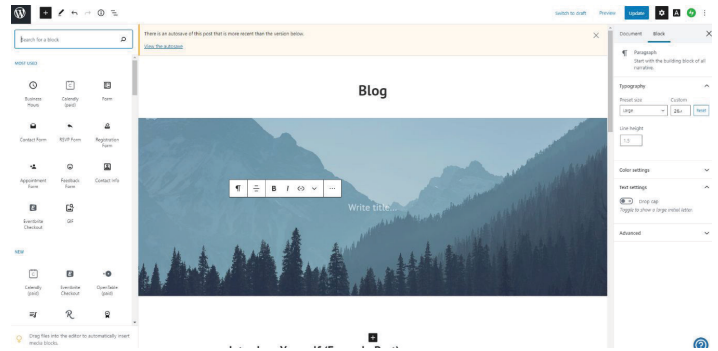


Figure 3. Customising a page.
Credits: WordPress (<https://wordpress.com/>)

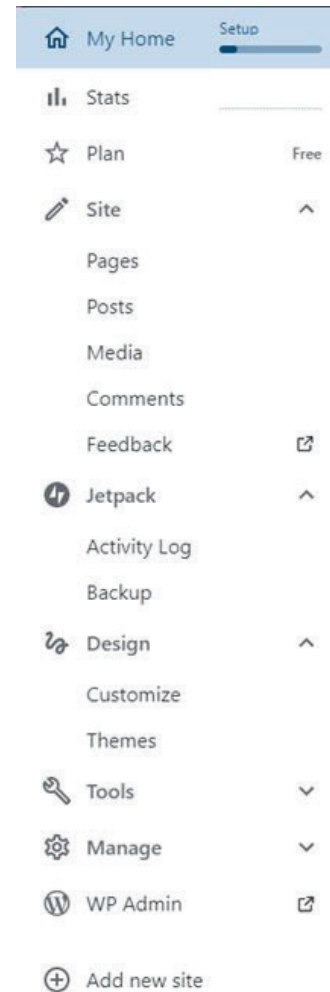


Figure 4. Adding a new page.
Credits: WordPress (<https://wordpress.com/>)

5. Using Mobile App Plugins—WordPress has plugins that can convert a web application into a mobile application. Once the website is ready, the plugins can be used from WordPress's toolbox to generate a mobile application, for both iOS and Android systems, from any website. These can be made public on Google Play Store and Apple App Store. For more information on these types of plugins, see this blog post on WPBeginner: <https://www.wpbeginner.com/showcase/best-plugins-to-convert-wordpress-into-mobile-app>.

6. Publishing the Website—The last step after adding all the details is to publish the website. Publishing a website means it is ready to go live for the world and will be public on the internet. For this, a domain name is required, as it is a unique identifier for your website (e.g., Google has google.com as the domain name). The domain name can be bought using WordPress or any other website. Details of that domain name can be filled in the form while publishing the website.

The following are the steps to register a domain name:

1. Go to any domain registrar website, like GoDaddy, Register, or Hostinger. Search for the desired domain name for your website using the search bar.
2. The price for that domain will be listed. Usually, domain names with “.com” subdomain are costly, and if your application name is prevalent like “agriculture.com” or “nitrogen.com,” then it will probably be in use by someone already. If that is the case, buying from the owner may be an option. Another option is choosing a different name or the same name except with a different suffix like “.co,” “.us,” or “.io” as a subdomain (e.g., greenandblue-farms.io).
3. If the search shows the availability of the domain, then it will also show the price associated with it.
4. Buying the domain is like online shopping. Add the domain to the cart and then pay via credit card or any other means available.

It is important to note that the domain name is just a name. This name needs to point to the web application. Without the domain name, the website will be inaccessible, even though the web application has an IP address.

Once you have bought your domain, you will be asked to point your domain to the IP Address where your website is hosted (resides). The IP address can be found in WordPress. Once this mapping is complete, your web application is ready for the world.

Summary

Using agro-applications helps in day-to-day work for farmers and researchers, but at the same time, developing them using web development tools like Java, C# ASP.NET, Node.js, and React can be hard for a person with no background in coding. CMS comes to the rescue as it provides easy development compared to coding an app. CMS is not very

flexible, except when creating a customized security CMS plugin, which requires coding skills. An application built using CMS is not recommended in the following scenarios:

- When large data processing or data analysis is required in the application
- When the application is based on some patented technology
- When security is a major concern
- When the business logic is regularly modified or the business logic is confidential
- When data is confidential

Although the above scenarios can be handled using CMS plugins, this would require either 1) using a pre-built plugin, which might pose a security concern as it is developed by a third-party, or 2) developing a CMS plugin in-house, which would require programming knowledge. It is recommended that in such cases, the app should be built from scratch. The app’s use case is one of the deciding factors that can help in choosing between CMS and developing an application using coding techniques.

References

- Csótó, M. 2015. “Mobile Devices in Agriculture: Attracting New Audiences or Serving the Tech Savvy?” *Journal of Agricultural Informatics* 6 (3): 75–84. <https://doi.org/10.17700/jai.2015.6.3.227>
- Fraisse, C. W., N. Perez, and J. H. Andreis. 2015. “Smart Strawberry Advisory System for Mobile Devices: AE516/AE516, 8/2015.” *EDIS* 2016 (1). <https://doi.org/10.32473/edis-ae516-2015>
- Franzen, D. 2018. “North Dakota Corn Potassium Recommendation Calculator.” NDSU. https://www.ndsu.edu/pubweb/soils/corn_k/
- Lariviere, M. 2011. “Kaizen and Cloud Computing for Agriculture.” *The Operations Room*. <https://operationsroom.wordpress.com/2011/01/20/kaizen-and-cloud-computing-for-agriculture/>
- Mbabazi, D., K. W. Migliaccio, J. H. Creane, J. H. Debastiani Andreis, C. Fraisse, L. Zotarelli, and K. T. Morgan. 2015. “Smart Irrigation Avocado App: A Step-by-Step Guide: AE513/AE513, 5/2015.” *EDIS* 2015 (4). <https://doi.org/10.32473/edis-ae513-2015>

Nelson, M. 2019. “Farmers’ growing reliance on technology highlights need for robust digital toolbox.” *The American Farm Bureau Federation*. <https://web.archive.org/web/20221215134706/https://www.fb.org/market-intel/farmers-growing-reliance-on-technology-highlights-need-for-robust-digital-t>

Prasad, S., A. K. Peddoju, and D. Gosh. 2013. “AgroMobile: A Cloud-Based Framework for Agriculturists on Mobile Platform.” *International Journal of Advanced Science and Technology* 59: 41–52. <https://doi.org/10.14257/ijast.2013.59.04>

Sawyer, J. E. 2019. “Springtime Precipitation – A Tool for Estimating Nitrogen Application Need for Corn.” *Integrated Crop Management News*. Ames: Iowa State University Extension and Outreach. <https://crops.extension.iastate.edu/cropnews/2019/06/springtime-precipitation-%E2%80%93-tool-estimating-nitrogen-application-need-corn>

Sawyer, J. E., and E. D. Nafziger. 2005. “Regional Approach to Making Nitrogen Fertilizer Rate Decisions for Corn.” In *35th North Central Extension-Industry Soil Fertility Conference Proceedings*, Vol. 21: Des Moines, IA. <https://dr.lib.iastate.edu/handle/20.500.12876/4376>

WordPress. n.d. “Website Builder.” Accessed January 23, 2024. <https://wordpress.com/website-builder/>

WPBeginner. 2024. “15 Best and Most Popular CMS Platforms in 2024.” January 2, 2024. <https://www.wpbeginner.com/showcase/best-cms-platforms-compared/>

Yevtushenko, O. 2016. “Types of Apps and Categories of Mobile Applications.” *Think Mobiles* (blog). Archived on March 13, 2017. <https://web.archive.org/web/20170313004608/https://thinkmobiles.com/blog/popular-types-of-apps/>