Developing Applications for Businesses using Low-Code Platform

A Technical Report submitted to the Department of Computer Science

Presented to the Faculty of the School of Engineering and Applied Science

University of Virginia • Charlottesville, Virginia

In Partial Fulfillment of the Requirements for the Degree

Bachelor of Science, School of Engineering

Krish Kothari

Spring, 2024

On my honor as a University Student, I have neither given nor received unauthorized aid on this assignment as defined by the Honor Guidelines for Thesis-Related Assignments

Briana Morrison, Department of Computer Science

Developing Applications for Businesses using Low-Code Platform

CS4991 Capstone Report, 2024

Krish Kothari
Computer Science
The University of Virginia
School of Engineering and Applied Science
Charlottesville, Virginia USA
ewq2yy@virginia.edu

ABSTRACT

In the 21st century, almost everything is done through online applications, and as businesses grow there is a need for accelerated application development to expedite and deploy business applications. Utilizing Appian's low-code platform automates and accelerates development of business applications, allowing for the creation of applications in weeks or months rather than years. I utilized Appian's easy-to-use interface and other tools to develop multiple applications throughout my time as an App Development Intern with Appian Corporation. The outcome was the successful delivery of a career fair application for the UVA career center. This application was scalable, adaptable, and showed the power of low-code platform by significantly improving the business process of the career center. Appian's low-code platform needs to be more flexible and allow for a better integration of complex GUI and AI models.

1. INTRODUCTION

The demand for rapid application development is at an all-time high. Business success is often dictated by the ease of access to the business's products and resources. With businesses changing their model and having a dynamic environment, it is crucial to have fast turnaround times in creating and updating their applications. The traditional approach of creating applications takes too long and is outdated, since coding every single aspect of an application does not meet the turnaround time required by the business and businesses cannot wait months or years to deploy important applications. This calls for an alternative development methodology that will ensure that applications are deployed and constructed with speed, efficiency and high quality.

The realm of low-code platforms is the perfect solution to this problem. The low-code platform is a development environment that can be used to create fully functional and complex applications through a Graphical User Interface (GUI). Appian Corporation is at the forefront and has created its low-code platform called Appian, which offers a solution that reduces development time remarkably. An application that might take months to code can take a week or even a few days with Appian.

2. RELATED WORKS

A review of scholarly sources that focus on the benefits of the low-code platform and creating applications with it shows similar work and explains why I chose to leverage Appians' low-code platform to build my application. These sources highlight the benefits of the low-code platform. Chatterjee (2023) discusses benefits, key components and features, and challenges and limitations of the low-code platform [1]. He talks about various benefits which made me realize just

how powerful the platform is: streamlined updates/maintenance, agility, scalability, flexibility, and many more. These concepts influenced me to learn the low-code platform and utilize it in my project.

Shah (2020) presents a similar use of the low-code platform [2]. Valley Health System (VHS) leverages the low-code platform to create mobile applications to enhance user experience. They can create critical business applications that leverage real-time data to optimize workflows in just ten weeks. This is very similar to my project; however, they solely focus on mobile applications whereas I focused my project on web applications. This use case of the platform truly shows its versatility because they used it for a healthcare app, whereas I used it for the career center.

3. PROJECT DESIGN

Our project pipeline was very simple because we utilized Appian's low-code platform, which meant we did not have to worry about using a specific tech stack. All the work we did was in the Appian low-code platform. We utilized agile development methodology to ensure a fast, smooth, and scalable application process. Also, my team used the Jira board to allow allocation of work tickets and work to be spread out across the team.

3.1 Review of System Architecture

The low-code platform, provided to my team by Appian Corporation was used as a foundation for rapid application development. The low-code platform's architecture is created to ensure maximum scalability. The platform offered a modular approach which made sure that requirements of the UVA career center were accounted for. The platform had an integrated sandbox feature in which my team and I could play around with different features before directly implementing them. The user-friendly visual development environment allowed for the creation of reusable application components. The system architecture authorized my team to incorporate business logic such as SEO-friendly webpages and data management by examining the databases to create a functional application. This architecture was crucial in making the development process scalable, adaptable, and agile. This architecture demonstrated the efficiency of low-code platforms.

3.2 Requirements

3.2.1 Client Needs

During our first week of the internship, we met with the UVA Career Center team to consult and figure out their specific needs. Our client, UVA Career Center had several requirements and some nice-to-have features for the application. We identified several key objectives. For example, they wanted to increase career fair attendance, streamline the registration process, and improve the efficiency of employer-student interactions. The big picture was to streamline the whole process for career fairs. From the registration to the career fair itself. Moreover, the client mentioned that the application should be dynamic. It should be able to be changed for every career fair with ease. These needs guided our development priorities and ensured that we create a scalable application.

3.2.2 System Limitations

Even though the low-code platform has extensive capabilities and features that make the platform the future of development, we encountered some crucial limitations. In any application, the GUI(Graphical User Interface) is very important because that is what attracts users. A limitation that we faced was incorporating complex GUI into the application. The client wanted to have a map of some sort that would guide students to specific booths with specific employers. We were not able to create a map because the

low-code platform does not support that feature. Moreover, with the low-code platform, we were only able to create five webpages which limited the amount of content we could put in the application. These limitations had a big effect on the scope of our project.

3.3 Key Components

Some general components we designed and included in the application were a portal system for employers and students, feedback component, and past/future events component. These components ensured a user-friendly experience for the user and ensured that they have important features at their fingertips.

3.3.1 Specifications

The application was designed using the platform's design elements such as forms, workflows, and data management tools. We utilized the platform card feature to make the base component of our web pages. We integrated external APIs to enhance the application's functionality. This allowed each student to have access to real-time data. We made a portal system for students and employers that allowed user data to be secured. Moreover, we made a separate component that was strictly for the UVA Career Center team so they could upload, delete, and update employer information for every career fair. Last, we made a component for the employers so they can add information such as job openings, recruiter information, and company information.

3.3.2 Challenges

One of the toughest challenges we encountered was overcoming the limitation of the platform's GUI capabilities. We needed to figure out a way to have a map UI so it is easy for the user to navigate the career fairs.

3.3.3 Solution

To address the challenge of the map in UI, we utilized the platform's ability to make unique grid layouts. We used the grid layout to make our version of the career fair layout. The UVA career center informed us that this application will be used for career fairs in the Scott Stadium and Newcomb Ballroom. So we designed a custom grid layout to replicate the layout of each career fair locations.

4. RESULTS

The result of this project was that we presented this to our clients, and they loved it. The application has yet to be launched; however, the UVA career center team was very ecstatic to have an option that streamlines the career fairs. For example, this application will ensure that there is no paper trail because all the relevant information is in the application. Lastly, we presented this application to our mentors at Appian and they were proud of the work we did in such a short time with a new technology.

5. CONCLUSION

My project demonstrated the sheer potential and power of low-code platforms. By leveraging Appian's low-code platform, my team and I were able to develop an application for the UVA Career Center in just a short period of time. We utilized the low-code platforms user-friendly visual environment and pre-built components to rapidly develop .

Our application was a highly scalable and adaptable application that significantly streamlined the career fair process for students, employers, and career center staff. Some key features included different portals that was tailored to each user group, real-time data integration, customized layouts to replicate different career fair venues, and administrative components for managing employer information across different events.

While utilizing the low-code platform enabled my team to have a quick turnaround

time, we also encountered limitations. Some of these limitations were advanced UI capabilities like interactive mapping and GUI. To combat this issue, we created solutions like a custom grid-based layout. This helped mitigate UI challenges.

Lastly, the project highlighted how the low-code platform provides a valuable option for accelerating business application delivery without sacrificing functionality and quality. It's an approach that is well-suited to modern organizational environments which require rapid application and scalabiliyty.

6. FUTURE WORK

The project has been completed and demoed to the career center team. The next steps are to gain real-user feedback and extend its capabilities. This project can be extended to make it a universal platform for all career center events. Moreover, the application can utilize the handshake API so the application can be unified. Moreover, Appian's AI components can be used to curate specific and tailored job recommendations based on student profiles and automated resume screening.

REFERENCES

[1] Shah, M. (2020, September 29). Valley Health System adopts Wavemaker Low Code to power unique mobile app delivery model and build critical business apps in Ten Weeks. Business Wire. https://www.businesswire.com/news/home/20200929005273/en/Valley-Health-System-Adopts-WaveMaker-Low-Code-to-Power-Unique-Mobile-App-Delivery-Model-and-Build-Critical-Business-Apps-in-Ten-Weeks

[2] Chatterjee, S. (2023, August 7). Comprehensive guide to low-code development in 2023. BrowserStack. https://www.browserstack.com/guide/low-code-development