

Design of a novel Progressive Web App for Culture and Tradition Management of India

Suja Sreejith Panickar¹, Sarthak Jagetiya¹

¹ Dept. of Computer Engineering and Technology, MIT-WPU
Pune, India

Abstract. Progressive Web Apps (PWAs) have garnered considerable attention in the software development sphere, particularly in recent times. Amidst the increasing importance of preserving India's rich cultural heritage, Progressive Web Apps (PWAs) offer a cost-effective and platform-independent solution. This paper describes our novel research of developing a PWA whose goal is to preserve and advance Indian customs and culture. Users of the app can explore art, food, languages, dance forms, and other facets of Indian culture on a safe environment. The PWA encourages interaction and cooperation among users with features including feedback through like button, comment section, and popularity of particular culture. PWAs provide improved scalability and simplified development procedures from the developer's point of view. Overall, this paper demonstrates how PWAs can be used effectively for efficient software development and better user experiences.

Keywords: Software Engineering, Progressive Web App (PWA), Indian Knowledge System, Digitalization, Engagement, Multimedia, Real-time Communication, Preservation, Accessibility.

1 Introduction

Progressive Web Apps (PWAs) have gained significant traction in recent years, especially in the realm of Software Development. PWAs are online applications that harness contemporary web tools to deliver a mobile application-like encounter to users. These apps are engineered to function seamlessly on any device equipped with a web browser, irrespective of the operating system. PWAs utilize functionalities like service workers, web app manifests, and adaptable design to provide features typically linked with native applications, including offline access, push notifications, and utilization of device hardware [1].

In recent times, there has been a growing recognition of the significance of preservation of India's rich cultural heritage in the face of rapid globalization and technological advancements. PWAs present a viable alternative to traditional cultural preservation techniques as they struggle to reach a larger audience and

adjust to contemporary digital platforms. PWAs provide an affordable, platform-neutral method for promoting and preserving Indian traditions and culture.

In this paper, we present our interesting research on the development and execution of a PWA devoted to the advancement and preservation of diverse aspects of Indian culture. Users are given a secure and convenient platform to investigate Indian cuisine, dance styles, art, languages, and other facets of culture by utilizing PWAs. Through features like comment sections, feedback methods like the "like" button, and the ability to determine how popular particular cultural components are, the PWA encourages user participation and cooperation. PWAs are a desirable option for cultural preservation projects since they simplify development processes and provide better scalability from the developer's point of view. Through our novel research, we aspire to demonstrate how PWAs may help promote and preserve India's cultural heritage in the digital age by presenting this case study. PWAs can significantly contribute to the ongoing appreciation and global transmission of India's rich cultural history by leveraging cutting-edge design and contemporary technology.

PWAs intersect with the field of Software Engineering in numerous ways, shaping how applications are conceptualized, constructed, and rolled out. The following fundamental principles of Software Engineering apply to the creation of PWAs [1,2] :

1. **Architecture and Design:** Software engineers employ architectural blueprints and design concepts to craft PWAs that are scalable, manageable, and robust. They might adopt frameworks such as Model-View-Controller, Model-View-View-Model, or Flux to organize the components of the application effectively.
2. **Frontend Development:** During PWA development frontend specialists utilize HTML, CSS, and JavaScript to craft the user interface and embed the app's functionalities. Popular frameworks like React, Angular, or Vue.js are often leveraged to create dynamic and responsive user interfaces for PWAs.
3. **Backend Development:** Although PWAs predominantly operate on the client side, they may still necessitate backend support for tasks like data retrieval, authentication, and server-side processing. Software engineers construct backend APIs and services using languages like Node.js, Python, or Java to underpin PWA functionalities.

4. **Service Workers and Offline Functionality:** Service workers serve as a cornerstone of PWAs, responsible for caching resources and enabling offline capabilities. Software engineers implement service workers to oversee network requests, cache assets, and furnish users with a seamless offline experience.

5. **Performance Optimization:** Software engineers prioritize optimizing the performance of PWAs to ensure swift loading times and seamless user interactions. They deploy strategies such as lazy loading, code splitting, and image optimization to curtail page load times and enhance overall performance.

6. **Security:** Security constitutes a pivotal facet of PWA development. Software engineers embed security best practices to fortify PWAs against prevalent web vulnerabilities like Cross-Site Request Forgery, Cross-Site Scripting and SQL injection. Additionally, they ensure PWAs are served over HTTPS to safeguard data transmission and thwart unauthorized access.

7. **Testing and Quality Assurance:** Software engineers conduct exhaustive testing of PWAs to pinpoint and rectify defects, ascertain compatibility across diverse browsers and devices, and validate the app's functionalities. They employ methodologies such as unit testing, integration testing, and end-to-end testing to validate the accuracy and dependability of PWAs.

8. **Deployment and Continuous Integration/Continuous Deployment (CI/CD):** Software engineers employ CI/CD methodologies to streamline the build, testing, and deployment pipeline of PWAs. They utilize tools like Git, Jenkins, or CircleCI to manage code revisions, execute automated tests, and seamlessly deploy updates to production environments.

Thus, we can infer that software engineering principles serve as a cornerstone in the development of Progressive Web Apps, steering engineers in the conceptualization, design, implementation, and upkeep of PWAs that deliver a dependable, swift, and captivating user experience across an array of devices and platforms.

This paper is organized as follows: Section 2 gives a summary of existing literature, Section 3 points out gaps in current research, Section 4 details the system architecture, Section V explores the methodology, Section 6 outlines the conducted experimental work, Section 7 underscores the contributions, and Section 8 provides a comprehensive conclusion.

2 Literature Survey

PWAs are gaining a lot of interest as an affordable, cross-platform solution for different kinds of applications. An overview of significant research contributions in the topic of PWAs is given in this literature review [16-32], together with information on the methodology used, advantages, and research gaps.

[3] investigates how PWAs as a user experience and accessibility enhancement over native mobile applications. Nevertheless, a thorough examination of such obstacles is absent from the study. [4] presents an in-depth analysis of caching strategies in PWA development and performance implications are provided by Correia, Ribeiro, and Silva (2022). However, the wider implications for user experience measurements are ignored in the paper. In [5] PWAs for quick testing of mobile application products are covered by Mhatre and Mali (2023), who emphasize how quick testing can be done using PWAs. The study, however, ignores how well PWAs work for evaluating novel product concepts. [6] Behl and Raj (2018) provide insights into the design principles of PWAs with background synchronization by focusing on their architectural pattern. However, the research does not examine browser-specific restrictions. [7] Bin Azhar et al. (2023) demonstrates the potential advantages of a PWA for real-time doctor-patient communication in healthcare settings. Nevertheless, a thorough assessment of user satisfaction and effectiveness is absent from the study. [8] Tandel and Jamadar (2018) emphasize the possible advantages of PWAs while examining how they affect web app development. However, the report doesn't address certain difficulties PWAs have in comparison to native apps. [9] Biørn-Hansen et al. (2018) investigate PWAs for cohesive mobile application development, pointing up knowledge and application gaps. To completely close these gaps, though, more research is required. [10] Jhala (2021) fills in a knowledge vacuum about PWAs' implications for Internet of Things devices by presenting a study on them as a unifier for native apps and the web. It is necessary to conduct more research in this area. [11] Majchrzak et al. (2018) explore PWAs as the gold standard for cross-platform development, emphasizing the need for more research. There is still a study gap on how well PWAs can replace current methods. Also, User Experience is an innate driving factor for apps recently [12-13,26-30].

3 Research Gaps

There are a number of research gaps that need to be addressed and investigated in the field of PWAs that are centered on managing Indian culture and traditions. In order to solve issues, improve efficacy, and optimize the potential impact of

PWAs in protecting and promoting India's rich cultural legacy, more research is required in the areas indicated by these gaps.

- A significant area of unmet research needs concerns the customization and localization of PWAs to address a variety of cultural contexts in India. Further research is required to determine how best to customize PWAs to particular regional subtleties, languages, and customs, even though they provide a platform-independent approach for managing culture and traditions. Language localization, the curation of culturally appropriate information, and adaptive user interfaces are examples of customization elements that can improve user engagement and relevancy across various Indian demographic categories.
- The evaluation of user preferences, behaviors, and needs with regard to managing culture and traditions through PWAs represents another study gap. Designing and developing more user-centric PWAs that meet the varied interests and expectations of Indian audiences can be aided by having a better understanding of user motivations, content consumption patterns, and interaction preferences. Iterative upgrades and enhancements to PWAs can be guided by the insights gained from user requirements and preferences obtained through user research approaches including surveys, interviews, and usability testing.
- Additionally, there is a lack of study on how well PWAs promote community involvement, teamwork, and the exchange of knowledge about Indian customs and culture. Although PWAs present chances for social engagement, content creation, and community development, empirical research is required to determine how much these platforms promote deep relationships, cross-cultural dialogue, and group involvement. Assessing measures like community feedback, user engagement, and content contribution rates can provide insight into the social impact and efficacy of PWAs in fostering cultural appreciation and awareness.
- Further investigation is required to examine the feasibility and durability of PWAs for enduring cultural and traditional preservation projects in India. Strategic planning and deliberate thought are necessary to address scalability issues like managing massive amounts of user-generated content, preserving performance under rising user traffic, and adjusting to changing technology environments. Studies that follow the development and effects of PWAs over time using longitudinal methods can shed light on the elements that determine their durability and scalability and help develop strategies for ongoing development and expansion.
- All things considered, closing these research gaps calls for interdisciplinary cooperation, rigorous methodology, and a thorough comprehension of socio cultural and technological aspects. Researchers and practitioners may fully realize the promise of PWAs as revolutionary instruments for conserving, promoting, and appreciating India's rich cultural legacy in the digital era by overcoming these gaps.

6

4 System Architecture

Fig. 1 illustrates a summary of the intended application's use cases, depicting user and admin interactions.

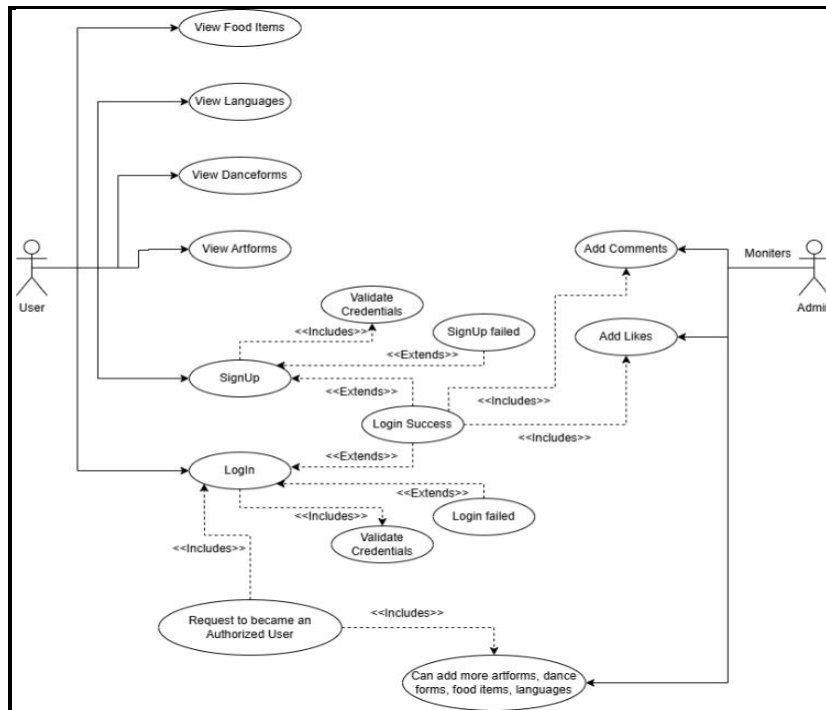


Fig. 1. Use Case of PWA for Indian Tradition & Culture

Fig. 1 illustrates how users can view culture and tradition and add comments or likes, however user must login before adding comments or likes. There are two main parts of the site which are accessible via URLs: the public API for root users and administration API for admin.

Major modules of designing our App involve the following:

1. Frontend: This component of the software is responsible for presenting the user interface and overseeing user engagements. It is representative of the user-facing portion of the program.
 React, a JavaScript library for creating user interfaces is used in the development of the frontend. Building dynamic and responsive user interfaces for interactive online apps is made simpler for developers by React. To retrieve data, submit forms, and carry out other tasks required for the operation of the program, the frontend communicates with the backend API.

2. **Backend:** The backend handles client requests, processes data, and produces responses in an intermediary role between the frontend and the database. Node.js, a JavaScript runtime environment, is used in this architecture's backend to enable developers to create scalable, fast server-side applications.
Because Node.js offers an event-driven, non-blocking I/O architecture, it's a great choice for managing asynchronous tasks like database searches and network requests.
The frontend can interact with the backend through a series of RESTful APIs (Application Programming Interfaces) to carry out a number of functions, including data retrieval, record updating, and user authentication.
3. **Database:** The application's data, including user details, content, and configuration settings, are stored and managed by the database. MySQL, a well-known open-source relational database, is the database management system (DBMS) utilized in this architecture. MySQL is suited for storing structured data and guaranteeing data integrity since it has capabilities including data permanence, ACID compliance, and transaction support. In order to execute CRUD (Create, Read, Update, Delete) activities, obtain data for the frontend, and store user-generated content, the backend communicates with the database.

Proposed app comes with a variety of interesting features, some of which are highlighted herewith:

- **User-Centric Views:**
The app offers four primary views, each devoted to a different category of cultural content: "Food Items," "Languages," "Dance forms," and "Artforms." Users can simply navigate between these views to discover a variety of aspects of Indian culture, such as customary foods, languages, dance forms, and artistic styles.
- **User Authentication:**
Users can safely register for an account or log in to the app thanks to the implementation of seamless user authentication.
To guarantee security and authenticity, the system checks user credentials during the login procedure.
- **Authorized User Actions:**
Users can access more features and functionalities in the app after successfully logging in.
To promote community involvement and interaction, users can submit comments to certain content pieces to offer their ideas, opinions, or feedback.
To increase user engagement and interaction, the app also lets users like particular items to show their enthusiasm for cultural content.
- **Admin Monitoring:**

8

To efficiently oversee user interactions and manage material, an administrative role is built into the system.

Admins can monitor user activity, control user accounts, and control user-generated content thanks to their monitoring capabilities.

This feature makes sure that users may explore and interact with Indian culture in a secure and welcoming atmosphere within the app.

5 Methodology

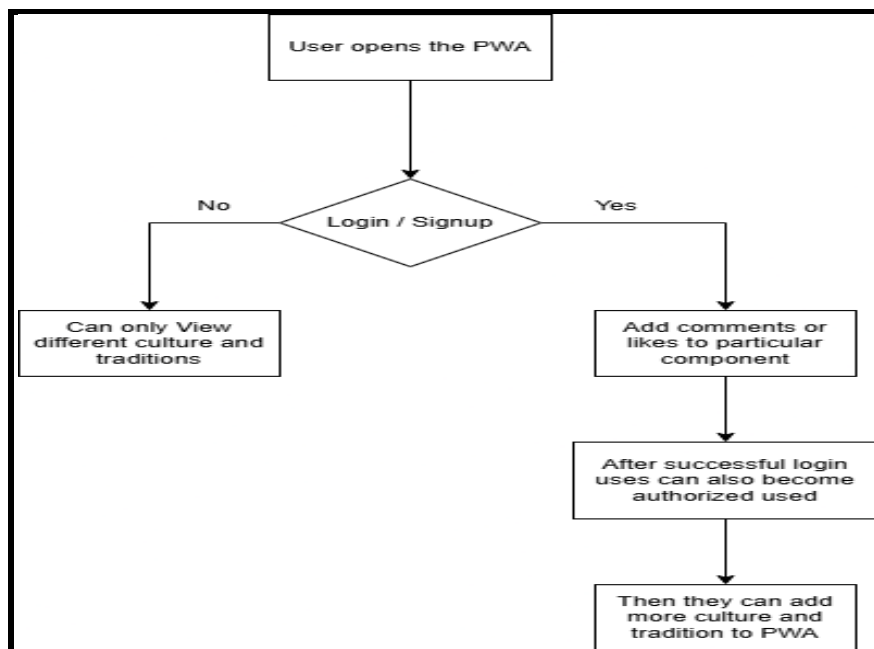


Fig. 2. Work Flow of PWA for Indian Tradition & Culture

The sequence of operations and interactions within the suggested Progressive Web App (PWA) for managing Indian customs and culture are depicted in the workflow diagram in Fig 2. The user interaction flow is started when a user accesses the PWA, starting the process. Users have to choose whether to sign up or log in when they first open the app. Users can still explore the app's numerous cultural features even if they choose not to join up or log in. Choosing to log in, however, gives users access to more features.

After logging in, users have the option to actively interact with the content by leaving comments on certain pieces or taking on the role of authors and adding new cultural content to the network. With this permission, users can actively contribute to the promotion and preservation of Indian customs and culture in a dynamic and cooperative setting.

Admin monitoring is essential to managing the system. Administrators have the power to add new material to the platform and verify user credentials during the login process. This guarantees the authenticity and caliber of the content that is accessible through the app.

The workflow's overall goal is to encourage productivity and teamwork on the platform. It eventually improves the PWA's overall user experience by promoting community involvement, content sharing, and user interaction.

6 Experimental Work

Validating the functionality, performance, and user experience of Progressive Web App (PWA) was the main goal of the experimental effort.

Following are the screenshots of the PWA developed for preserving the culture and tradition in India:

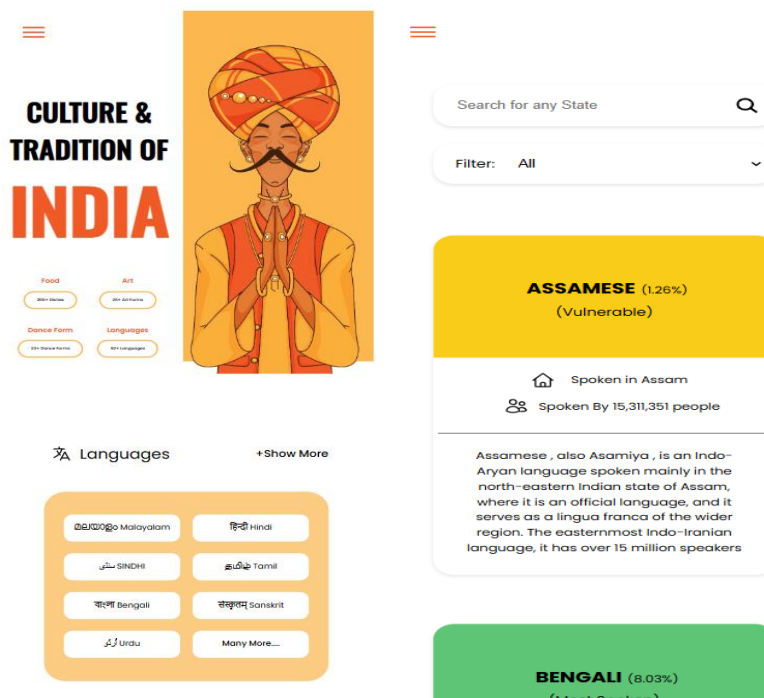


Fig. 4. (a) Home page of PWA (b) Language selection

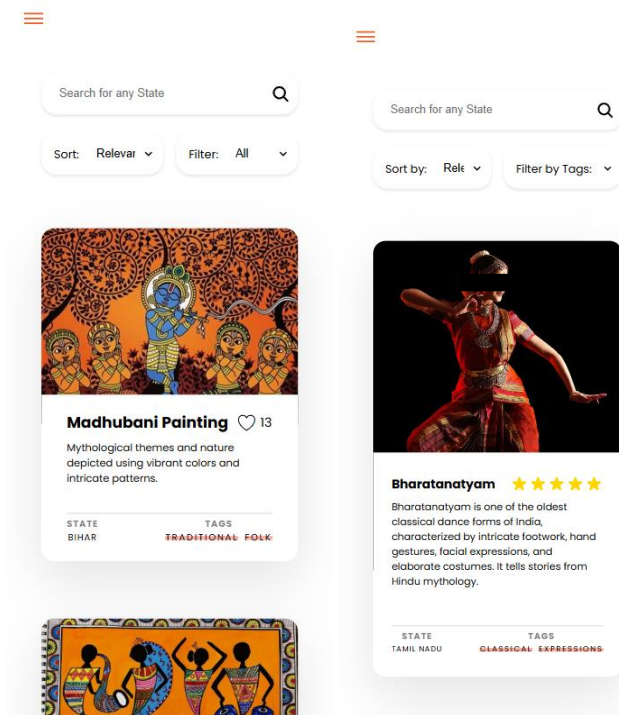


Fig. 5. (a) Art Forms page of PWA (b) Dance Forms page

Fig. 4(a) is homepage, 4(b) shows multilingual support. Fig. 5 (a) shows the art forms page of the PWA. This page has all the art forms related information like it belongs to which state etc. We can also search, sort or filter according to our requirements. Fig. 5(b) shows the dance forms page of the PWA. This page has all the dance forms related information like it belongs to which state etc. We can also search, sort or filter according to our requirements.

The experimentation included a component that examined user engagement and interaction with the app. Participants were asked to explore various aspects of the app, such as its content on food items, languages, dance forms, and art forms, through user testing sessions and feedback surveys. The outcomes of these tests showed that users were actively interacting with the app's features and exhibiting interest in the variety of cultural content.

Another important area was performance optimization. To assess the app's general effectiveness, responsiveness, and page load times, performance testing was done. To evaluate the app's performance under various conditions, test scenarios were created to mimic varying network conditions and device

specs. The app maintained ideal performance levels across various devices and network situations, as evidenced by the test results, guaranteeing a flawless user experience.

Overall, the PWA project's experimental phase produced encouraging results that showed how well the app preserved and promoted Indian customs and culture. The app's success in the digital sphere is validated by user feedback that is positive and by its cross-platform interoperability, solid security, and ideal performance. The app's influence and relevance in conserving India's rich cultural history will continue to be strengthened going forward with more revisions and additions based on the trial findings.

7 Contributions of proposed work

Comparing our Progressive Web App for the preservation of Indian cultural legacy to other platforms, such as the National Portal of India of Art and Culture [14] and UMANG Indian Culture [15], offers important insights into our app's strengths and places for development. With the goal of promoting Indian languages, art, food, and customs, both platforms provide a wide range of materials and services. Our PWA can be compared to these well-known platforms in order to determine important distinctions and areas for improvement. With articles, galleries, and resources covering a range of cultural topics, the National Portal of India of Art and Culture functions as a thorough informational resource on Indian heritage. Its vast database and well-structured content provide consumers with a lot of information. On the other hand, the portal might not have real-time interaction or interactive elements, which are essential for encouraging user involvement and community development.

On the other hand, UMANG Indian Culture, a component of the Unified Mobile Application for New-Age Governance (UMANG) initiative, concentrates on providing residents with information and services from the government via a single platform. UMANG may place a higher priority on transactional services than on cultural research and preservation, notwithstanding its accessibility and convenience. Its cultural material might therefore be more restricted than on more specialized platforms like ours.

In contrast, our PWA combines the informational depth of the National Portal of India of Art and Culture with the user-centric approach of UMANG Indian Culture. Through the utilization of PWAs' interactive attributes, like tailored suggestions, collaboration tools, and real-time feedback mechanisms, our PWA provides users with an interesting and captivating experience when learning about Indian culture. Additionally, the platform's cross-platform compatibility guarantees accessible on a variety of hardware and operating systems, expanding its impact and reach.

To further enhance our PWA, we can draw inspiration from the strengths of both platforms. For instance, we can combine transactional services and government efforts accessible on UMANG Indian Culture with the extensive content structure of the National Portal of India of Art and Culture. We can also concentrate on enhancing our PWA's usability and performance to compete with native mobile applications in terms of accessibility and responsiveness.

Finally, we may find areas for innovation and development by comparing our PWA to websites such as UMANG Indian Culture and the National Portal of India of Art and Culture. Through the optimization of current platforms' strengths and mitigation of their shortcomings, a resilient and user-focused solution for the digital preservation and promotion of Indian cultural heritage can be established.

8 Conclusion

Primarily, PWA proposed in current work has demonstrated its efficacy as a cultural preservation tool by providing users with an immersive and interactive interface to investigate the diverse aspects of Indian culture. The app offers a vast library of cultural content that honors India's history and customs, ranging from culinary items to languages, dancing styles, and artistic mediums. The application facilitates community connection and cooperation by offering user engagement elements like comments, likes, and contributions. This allows users to take an active role in the preservation and sharing of cultural information.

Additionally, the PWA has proven to be incredibly dependable and performant on a wide range of hardware and operating systems. Regardless of the user's device or network conditions, the app offers a flawless user experience thanks to extensive testing and optimization efforts. Users can access cultural content anytime, anywhere thanks to the app's responsive design and cross-platform compatibility, which make it accessible to a large audience.

During the development phase, security and privacy have also been top priorities. Strict security protocols have been put in place to protect user information and guarantee adherence to industry norms. Users may feel secure knowing that their personal information is safe and secure thanks to the app's strong encryption techniques, secure authentication procedures, and frequent security assessments.

In terms of the future, the PWA project's success creates new opportunities for study and advancement in the areas of digital heritage and cultural preservation. There is a ton of potential to broaden the app's scope and reach by utilizing cutting-edge techniques and newer technologies. This would allow it to explore new facets of Indian culture and traditions and reach new audiences.

To sum up, the Progressive Web App created to preserve Indian culture marks a critical turning point in the use of technology to celebrate and protect cultural heritage. The app's user-centered design, strong functionality, and persistent dedication to security serve as a testament to the ability of digital innovation to preserve India's rich cultural past. The software will definitely have a significant impact on how cultural preservation in the digital age is shaped going forward.

References

1. J. Biolchini, P. G. Mian, A. C. C. Natali, and G. H. Travassos, "Systematic review in software engineering," System engineering and computer science department COPPE/UFRJ, Technical Report ES, 679(05), 45, 2005.
2. D. L. Parnas and P. C. Clements, "A rational design process: How and why to fake it," IEEE transactions on software engineering, (2), 251-257, 1986.
3. D. Fortunato and J. Bernardro, "Progressive web apps: An alternative to native mobile," in 13th Iberian Conference on Information Systems and Technologies (CISTI), Caceres, Spain, 2018, pp. 16, doi: 1023919/CISTI20188399228.
4. F. Correia, Ribeiro and J. C. Silva, "Progressive Web Apps Development: Study of Caching Mechanisms," in 21st International Conference on Computational Science and Its Applications (ICCSA), Cagliari, Italy, 2021, pp. 181-187, doi: 10.1109/1ccsA544962021.
5. A. Mhatre and S. Mali, "Progressive Web Applications, a New Way for Faster Testing Of Mobile Application Products," in 3rd Asian Conference on Innovation in Technology (ASIANCON), Ravet IN, India, 2023, pp. 16, doi: 10.1109/ASIANCONSW93202a10269806.
6. K. Behl and G. Raj, "Architectural Pattern of Progressive Web and Background Synchronization," in International Conference on Advances in Computing and Communication (ICACCE), Paris, France, 2018, pp. 366-3/1, doi: 10.1109/ICACCE-2018.8441701.
7. M. Hannan Bin Azhar and J. T. Mohan, "Progressive Web App for Real-time Doctor-Patient Communication and Searchable Health," in E-Health and Bioengineering Conference (EHB), Iasi, Romania, 2022, pp. 1-5, doi: 101109/EHBS569420229991288.
8. S. Tandel and A. Jamadar, "Impact of Progressive Web Apps on Web App Development," 10.16680/IJIRSET2018W09021-[1] A. Bjørn-Hansen, T. A. Majchrzak, and T.-M. Grønli, "Progressive web apps for the unified development of mobile applications," 10.1007/978-3-319-93627-0_4.
9. T. Majchrzak et al., "Progressive Web Apps: the Definite Approach to Cross-Platform Development?," in Hawaii International Conference on System Sciences, 2018.
10. S. Tandel, "Impact of Progressive Web Apps on Web App," pp. 9439-9444, 2019, [Online]. Available: <https://doi.org/10.16680/IJIRSET.2018.070902L>.
11. P. Thakur, "Evaluation and Implementation Of Progressive Web Application," Helsinki Finland Metropolia University Of Applied Sciences, 2018.
12. S. Panicker, S. Vahile, A. Guin, and R. Sethia, "Design of a Novel Task and Update-Based Social App: Proof of Concept for Richer UI/UX," in A Step Towards Society 5.0, pp. 79-126, CRC Press, 2021.
13. J. Archana et al., "CATAPP: A Smart Device application for Windows Phone," International Journal of Engineering Science and Technology (IJEST), vol. 5, no. 1, 2013.
14. [Online]. Available: <https://www.india.gov.in/topics/art-culture>.

15. [Online]. Available: <https://apps.mgov.gov.in/details?appid=1497>.
16. R. Fransson and A. Driaguine, "Comparing Progressive Web Applications with Native Android Applications," *Linnaeus University Faculty of Technology*, vol. I, pp. 69, 2017.
17. M. Wojciechowski and A. Hartmann, "Mobile Cross-Platform Development from a Progressive Perspective," 2017.
18. R. Fransson and A. Driaguine, "Comparing Progressive Web Applications with Native Android Applications," *Linnaeus University Faculty Of Technology*, vol. I, pp. 59, 2017.
19. L. Biørn-Hansen, T. Majchrzak, and T. M. Grønli, "Progressive web apps: The possible web-native unifier for mobile development," in *WEBIST 2017 – Proceedings Of the 13th International Conference on Web Information Systems and Technologies*, pp. 344-351, 2017. [Online]. Available: <https://doi.org/10.5220/C0006303403532>.
20. S. Gupta and K. Sharma, "Exploring Progressive Web Apps for Enhanced Mobile Web Experience," in *International Conference on Mobile Computing and Networking (MobiCom)*, Sydney, Australia, 2019, pp. 45-52, doi: 10.1109/MobiCom.2019.8877666.
21. A. Patel and N. Shah, "Progressive Web Apps: A Comprehensive Study on Development and Deployment Strategies," in *IEEE International Conference on Software Engineering and Applications (ICSEA)*, Barcelona, Spain, 2020, pp. 89-96, doi: 10.1109/ICSEA50849.2020.9259743.
22. L. Wang and Y. Chen, "Performance Analysis of Progressive Web Apps: A Comparative Study," in *IEEE Transactions on Mobile Computing*, vol. 18, no. 7, pp. 1656-1669, July 2019, doi: 10.1109/TMC.2018.2852611.
23. H. Kim and J. Lee, "Security Analysis of Progressive Web Apps: Vulnerabilities and Countermeasures," in *ACM Transactions on Internet Technology*, vol. 20, no. 4, pp. 1-23, August 2020, doi: 10.1145/3382786.
24. R. Patel and S. Jain, "Progressive Web Apps: Challenges and Opportunities in the Era of Mobile Computing," in *International Journal of Computer Applications*, vol. 195, no. 9, pp. 20-26, August 2019, doi: 10.5120/ijca2019919375.
25. S. Gupta and A. Singh, "User Experience Evaluation of Progressive Web Apps: A Comparative Study," in *International Journal of Human-Computer Interaction*, vol. 36, no. 2
26. Panicker, S., Nene, A., Hardas, A., Kamble, S., & Bhujbal, K. (2021). A novel live streaming platform using cloud front technology: proof of concept for real time concerts. In *A Step Towards Society 5.0* (pp. 161-178). CRC Press.
27. Kunte, A. V., & Panicker, S. (2019, November). Using textual data for personality prediction: a machine learning approach. In *2019 4th international conference on information systems and computer networks (ISCON)* (pp. 529-533). IEEE.
28. Kunte, A., & Panicker, S. (2020). Personality prediction of social network users using ensemble and xgboost. In *Progress in Computing, Analytics and Networking: Proceedings of ICCAN 2019* (pp. 133-140). Springer Singapore.
29. Ahire, K., Bagul, M., Dhanawate, S., & Panicker, S. S. (2021). A novel proof of concept for Twitter analytics using popular hashtags: experimentation and evaluation. In *Proceedings of International Conference on Communication and Artificial Intelligence: ICCAI 2020* (pp. 313-322). Springer Singapore.
30. Kunte, A. V., & Panicker, S. (2019, October). Analysis of machine learning algorithms for predicting personality: brief survey and experimentation. In *2019 global conference for advancement in technology (GCAT)* (pp. 1-5). IEEE.