**System implementation and maintenance**

**Implementation plan and Executive summary**

**Executive Summary**

This is a summary of the implementation plan as will be applied by Mince Company. The implementation plan covers the application, the development methodology applied, the integration approach, implementation strategy, the conversion strategy, implementation strategy, testing strategy and the knowledge/training strategy. The application is factored to run on all modern media browsers with a keen attention to responsiveness of the end user services screens.

The application has three interconnected component that include the end user interface, the backend and the servers. All the three components communicate via endpoints and APIs. Data that is generated on the application both by the system and the user shall be handled only as per the defined policies and terms and conditions of the use, no customer data shall be used without consent of the user.BI shall use the data generated by the system to predict eventualities and forecast sales changes. Unit tests and general tests shall be applied at hours that do not impact gross business performances and all the laid down communication protocols shall be applied to clients and affected parties before, during and after the impact.

Continuous knowledge improvement and training shall be done to align every team member on-board to familiarise themselves with the apps environment so that they can be able to sort client queries and enquiries as soon as they can.

**Implementation plan**

**The application.**

Crostini Mince application will run as web app. The application targets responsive access on all media browsers of all common resolutions amongst its users. The language chosen for the framework should include JavaScript for the frontend with material UI framework. The backend application should run on Node JavaScript platform. The application will consume Google firebase NOSQL database platforms.

Back to the user interface, the authentication mechanism for the different users shall be based on Google Oath and as again a service from the Firebase architecture. Third party services such as payment platforms, SMS integrations shall be applied where necessary. We plan to use Bank cards for checkouts and PayPal for customer’s service renewals on the app. We arrived at the above solutions because React Js is maintained at Facebook and so security is up to date, secondly, React Js gives an a faster development approached when it comes to project set up and components state management Aggarwal (2018). Node js as a backend service shall help in a quicker processing of JavaScript code and its implementation.

We consider the application to be developed in house because this will make it easier for the development team to easily manage changes and offer any technical requirement on the project whenever need be. Secondly, developing in house apps are much easier and cheaper to acquire, since there are no monthly charging fees required for implementation.

**Software development Methodology**

The software development approach shall consume the agile methodology approach. The program shall be broken down into phases where upon each the requirements of each component shall be studied and applied to each level of the development program. According to Srivastava et al (2017) The advantages of this methodology include the ability to determine risks before they occur, reduced bugs on the program application and the ability of continuous improvement since each project node component is treated as its own and can eventually be modified and improved.



Figure Agile methodology

**Integration Approach**

All application component shall be integrated together in a seamless communication protocol with each node being able to communicate with each other with no jamming. There are three major components that will be implemented and they include the User interface, the backend service and the service. This components will communicate to each other by use of the relevant end points and APIs.

Where third party services integration are needed, each node that requires the services shall have its components integrated as well, in this case, third party services may include SMS, payment systems or even authentication systems. Below is the likely scenario of the integrated application overview:

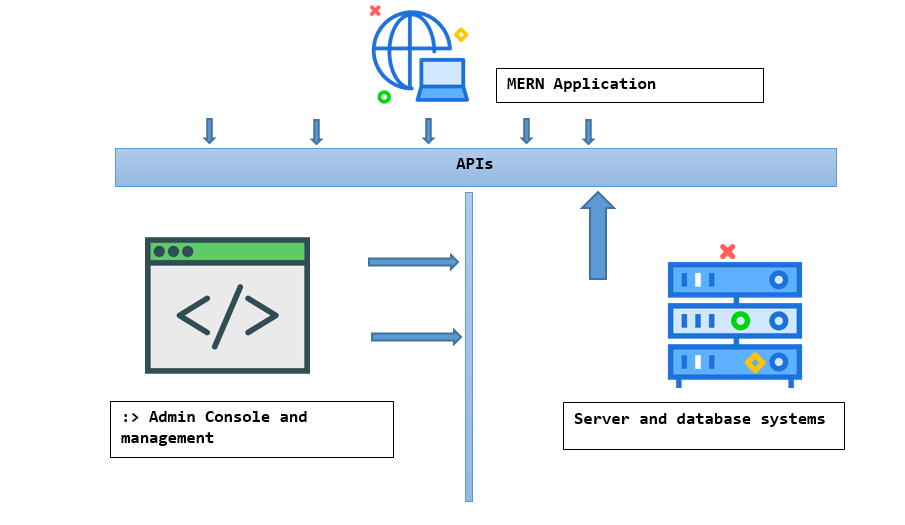


Figure application diagram components

**Implementation Strategy**

The finished application shall be deployed on a cloud infrastructure platform, the cloud could be third party such as AWS, Azure or GCP. The advantages of these strategies include safety and the frequency of maintenance so most of the time, the servers will always be up and running, will very minimal downtimes to affect user activities. For instance, cloud architects within the organization can use the Google cloud platform tool to deploy the app to the cloud servers and there after monitor its state from time to time.

There will also be a separation of IP address access, one will be a public domain URL for access to the site for all common end users and the other IP address masked to intranet subnets for internal access. This shall also improve the security of the application. After successfully deployment, necessary trainings should follow for the front end teams to familiarise themselves with the application even as services are targeted for release to the market.

**Conversion strategy**

Data handling will be at the core of the application. These include data generated by the end users and data that is system generated. Business intelligence analytics tools shall be applied where necessary to interpret the data collected to meaningful business decisions that will help drive the company forward to its mandate. Also, proper polices with the right terms and conditions shall be put in place to ensure that user data is not violated and breached, These policies, terms and conditions shall be publicly available to consumers and end users on the company website whenever users need to access them.

**Deployment strategy**

These are activities that encompass how the application will be pushed online for end user services. The first step is identification of the cloud services provider which shall be identified as selected based on cost, latency, availability and security. Once this is done, then cloud engineers shall familiarise themselves with the environment and make necessary configurators that are needed to implement the site.

The configuration by cloud engineers shall include databases, disk fragmentations, tests, severity patches and IP address definitions where necessary. The deployment should also be done at a time when services are very less likely to be interrupted so as to also allow time for the users to test and feel the changes.

**Testing strategy**

The testing process shall be an end to end activity. The tests shall be carried out on all hardware materials, software and network instances. The hardware being tested can include the personal computer memories, screen appearance and resolution changes to responsiveness. The application shall also be tested at different intervals of the day to check to monitor how traffic is likely to affect app performance, if all app components perform as required and none has a bug. Where bugs and difficulties are encountered while using the application, then hot fixes should be done before the application is released to the market Khanam et al (2018). Though if this happens when app is in production, changes may be redone on the branch app and staged. Once ready, a small windows of redeployment can be applied to the affect the normal user services with the right communication. An example of such communication script can be,

*“Dear customer, we are sorry that our services are currently inaccessible to due to a maintenance that is currently ongoing to better our services, kindly bear with us as we make this change, we shall resume shortly*”

Finally, in light of security, the tests should be done every specific number of days or months as defined by the management to ensure continuous service reliability and security.

**Knowledge transfer and training Plan**

This shall include the managing of the apps information to end users. The process included define who currently has knowledge about the system, These can include software engineers and application leads, who in turn can train first level management staff to then train and align their front end teams.

Along with are the proper system documentation plans that should be distributed to the application support teams to help them easier responds to client queries from within and out. If any change is made by the Engineer then same communication should also be passed out to the other team members so that they are aware. When staff are well trained, their confidence level and morale also spikes up Okechukwu (2108).

**REFERENCES**

Srivastava, A., Bhardwaj, S., & Saraswat, S. (2017, May). SCRUM model for agile methodology. In *2017 International Conference on Computing, Communication and Automation (ICCCA)* (pp. 864-869). IEEE.

Aggarwal, S. (2018). Modern web-development using reactjs. *International Journal of Recent Research Aspects*, *5*(1), 133-137.

Okechukwu, W. (2017). INFLUENCE OF TRAINING AND DEVELOPMENT, EMPLOYEE PERFORMANCE ON JOB SATISFACTION AMONG THE STAFF. *Journal of Technology Management and Business*, *4*(1).

Khanam, Z., & Ahsan, M. N. (2017). Evaluating the effectiveness of test driven development: advantages and pitfalls. *International Journal of Applied Engineering Research*, *12*(18), 7705-7716.