SOFTWARE ENGINEERING

QUALITY MANAGEMENT

## TABLE OF CONTENTS

[EXECUTIVE SUMMARY 2](#_TOC_250007)

[INTRODUCTION 3](#_TOC_250006)

[SOFTWARE QUALITY MANAGEMENT 3](#_TOC_250005)

[KEY APPROACHES TO QUALITY MANAGEMENT 3](#_TOC_250004)

[SOFTWARE QUALITY MANAGEMENT METRICS 4](#_TOC_250003)

[METRICS ANALYSIS 5](#_TOC_250002)

[CONCLUSION 7](#_TOC_250001)

[BIBLIOGRAPHY 8](#_TOC_250000)

# EXECUTIVE SUMMARY

## OctoFlex Technologies is a start-up IT company that specializes in creating custom software solutions for small and medium-sized businesses. team of expert software engineers and designers create high-quality software tailored to meet the unique needs of each client. OctoFlex Technologies is committed to staying at the forefront of the software development industry by placing a strong emphasis on innovation and quality.

### The Company

## OctoFlex Technologies has gained a reputation for producing high-quality software that is both reliable and cost-effective. Our software solutions are known for providing significant economic value to clients.

### The Market

## In the highly competitive software development industry, OctoFlex Technologies stands out by focusing on client satisfaction and delivering exceptional work. We work closely with clients to fully understand their requirements and deliver solutions that surpass their expectations.

# INTRODUCTION

OctoFlex Technologies aims to develop Blood Support+ Application. To ensure that the product we deliver is of high quality, we will use various key approaches that is effective and efficient to quality management. This report will outline the key approaches to quality and the metrics that will be measured to assess software quality.

# SOFTWARE QUALITY MANAGEMENT

Software quality management submits the modifications to the product development process to ensure that the appropriate level of quality is met which is acceptable. The team's culture is something that SQA seeks to build, and everyone is held accountable for it.

To achieve independence of cost and schedule adherences, software quality management should be separate from project management. Both the process quality and the product quality are immediately impacted.

# KEY APPROACHES TO QUALITY MANAGEMENT

* Continuous Enhancement: We will aim to employ continuous enhancement that will make sure that our processes are continually evolving to meet all the necessary customer needs. This approach will involve systematic regular reviews of all the processes, regular training of all the team members, and concentration on identifying and sorting out the issues as they arise.
* Adherence to Industry Standards: We will follow all the industry standards such as ISO/IEC 12207 standard which has been set up to make sure that our software development processes are consistent, efficient, and effective. We will use ISO 9000 as a fixed standard to make sure that we are meeting the requirements for quality assurance. We will also adhere to coding standards and best practices to make sure that the code is of high quality.
* Customer Review: We will actively seek customer feedback throughout the development process of our Blood Support+ Application. This will enable us to spot any problems as they arise and make the required adjustments to guarantee that our software fulfils the requirements and expectations of our clients.
* Quality Assurance: This is a methodical quality management procedure where we will decide whether a process, we do is going to satisfy our client's needs and standards. We will create and uphold the necessary requirements for Blood Support+ Application to ensure that the software we are relying on is quality assured. We will boost our client’s confidence to new heights with a suitable quality assurance system while also effectively improving the way we manage our job.
* Quality Planning: We will address several issues, such as the market our product (Blood Support+ Application) is intended to dominate, the timing of new updates, handling potential hazards, having quality objectives, and creating the necessary risk management strategy.
* Quality Control: Since there can be a cause for commotion if control processes aren't adjusted, agile quality management will be preferred by OctoFlex Technologies. One illustration is that we will make sure that we address any modifications from the prior test after reviewing the software. Software testing that has the necessary documentation to assist the testing process would be another illustration.

# SOFTWARE QUALITY MANAGEMENT METRICS

The three P's—product, process, and project attributes—are realised through metrics, which are merely numerical indicators or arrows. The characteristics of the product are intricacy, quality, and size. The process attributes discuss quality enhancement and rapid development. The productivity, quantity of resources, and expenditures would be included in the project attributes.

# METRICS ANALYSIS

To measure the quality of the Blood Support+ Application, we will use several key metrics which are:

* Defect Density: In order to determine the areas in which we need to make improvements, we will keep track of the number of flaws (defects) in our software and calculate the defect density.
* Measurement of code coverage will be done to make certain that our programme is adequately tested and that our tests are comprehensive.
* Customer Satisfaction: In order to make sure our software meets customer expectations; we will measure customer satisfaction using surveys and feedback forms.
* Cycle Period: Similar to lead time, cycle time is a metric that is measured in terms of tasks as opposed to user stories. The cycle time is the amount of time it takes to construct the database, whereas the lead time is the amount of time it takes to have your database totally ready. We will be use cycle time to predict delivery times for next sprints.
* Rate of Application Crashes: We will measure how frequently an environment experiences website or mobile app crashes. It is a sign of good coding when a the rate of application crashes are very low which translates to a stable software that won't crash.
* Group Velocity: Group Velocity is an extremely important metric. It will show the number of tasks or user stories that our team has finished during the first sprint. We will add the incomplete tasks to the backlog without involving the tasks. Only fully finished user stories will be considered when calculating the release velocity for our Blood Support+ Application.
* Number of errors per sprint: The amount of problems we will uncover during each sprint would be tracked by these metrics. Each sprint's user story quality may be evaluated using this metric.
* Rate of First-Time Passes: These metrics support the agile manifesto of rapid delivery and excellent quality. We will keep track of the number of test cases that can pass during the initial run. This would track the calibre of the development. This would imply that there were no bugs discovered in the code we wrote.

# CONCLUSION

In conclusion, the success of the Blood Support+ Application heavily relies on the quality management approaches employed during the development process. OctoFlex Technologies will make sure that the software is of high quality by using continuous enhancement, adherence to industry standards, customer reviews, quality assurance, quality planning, and quality control. To measure the quality of the software, OctoFlex Technologies will use metrics such as defect density, code coverage, customer satisfaction, cycle period, rate of application crashes, group velocity, number of errors per sprint, and rate of first-time passes. These metrics will enable us to identify the areas of required enhancement and ensure that the Blood Support+ Application meets the expectations of its customers. By using effective and efficient quality management approaches and metrics, OctoFlex Technologies is committed to delivering a high-quality Blood Support+ Application.

# BIBLIOGRAPHY

***730-2014 IEEE standard for software quality assurance processes***. (n.d.).

***ISO/IEC/IEEE 12207: 2017(E) first edition 2017-11: ISO/IEC/IEEE international standard - Systems and software engineering -- Software life cycle processes***. (2017).

***ISO/IEC/IEEE international standard - Software and systems engineering -- Software testing -- Part 5: Keyword-driven testing.*** (n.d.). https://doi.org/10.1109/ieeestd.2016.7750539

Pressman, R. S., & Bruce R. Maxim, D. (2014). ***Software engineering: A practitioner's approach***.

McGraw-Hill Education.

***Quality management***. (n.d.). Online Tutorials

Library. https://[www.tutorialspoint.com/software\_testing\_dictionary/quality\_management](http://www.tutorialspoint.com/software_testing_dictionary/quality_management)

.htm

***Software quality management: A comprehensive guide with examples and best practices***. (n.d.). Most Powerful Cross Browser Testing Tool Online |

LambdaTest. https://[www.lambdatest.com/learning-hub/quality-management#b](http://www.lambdatest.com/learning-hub/quality-management#b)