Final Java Project – Report
Java Project to check the BMI (Body Mass Index)/BMR (Basal Metabolic rate) in Humans and Animals

Team Members – Group 4

Sumanth Mohan - c0752365
Abhishek Rangra - c0755154
Dhaval Kumar - c0752581
Shubham Nanda - c075900

Executive Summary

In this particular project report procedure, a detailed discussion has been provided about the various steps of development of a software application. The required report also analyses specific approach methods of designing and developing a BMI/BMR calculation programming using core java concepts like inheritance and the use of loops where required. This particular program is separately designed for the calculation of Basal Metabolic Rate and Body Mass Index for Humans and Animals. This program takes three inputs height, weight and age to determine the two bodily conditions and presents them as output. An additional procedure to specify the nature of BMI value has also been assigned.

Table of Contents

Introduction

Software development life cycle is a system of processes and methods by which particular software is planned, designed and implemented. In this particular topic of discussion, design and implementation of specific BMI/BMR calculator are presented that takes separate values for animals and humans. In this particular program development, a choice has been provided for the participants on the type of checking that they want to do. A database system has been linked with the particular program using "JDBC" connection. Various core concepts of java have been implemented within the procedures. This particular program has been implemented without a GUI based output.

1. Aim and Objectives

This particular project report has been documented with the purpose of serving the basic requirement analysis and implementation of core java concepts within a particular application. The BMI or Body Mass Index formula has been implemented in order to receive input from the users regarding their weight and height, calculate the specific BMI and present it as output. Another program has been separately designed within the same main class of java coding to provide an output regarding the BMR or Basal Metabolic Rate of the users. Nazir & Nazir (2018, p. 251) have stated that this particular program is designed to take specific weight, height, and age from users, calculate the BMR and present the result as output.

Aim and Objective of this program are to create separate methods of checking for animals and humans. Various core java concepts like inheritance, polymorphism, use of loops, etc. have been presented in this coding design and successful results are to be provided at the time of testing. A method of choosing the particulars of this procedure has been designed into the

programming structure for proper ease of use (This project has been designed keeping in mind the particulars of Software Development Life Cycle and its various methodologies and concepts.

2. Software Development Life Cycle

Software Development life cycle is an important part of the software design and implementation. Along with the help of various SDLC models like the waterfall model or the spiral model, this particular methodology provides an overview of the various requirements of a software developer in order to create a software application for their concerned clients. Software development life Cycle contains various procedural steps of reference. These particular steps are mentioned below:

- ***Planning*** is the initial part of an SDLC. Planning requires that the development professional assess various immediate requirements for gathering of knowledge regarding the actual need for development of the software and work accordingly
- ***Requirement Analysis*** is part of SDLC where developer is required to analyze the various requirements specifically gathered for serving the exact need for software development
- ***Design and Implementation*** is the most necessary part of SDLC. This is a process where developers need to design the actual source code of the proposed software application in order to implement the various methods needed to serve specific software requirements with the help of programming languages and the use of Integrated Development Environment
- ***Testing*** is the final procedure of software observation and assessment before it is released for actual use. In this particular process software application and its various

functionalities are thoroughly tested for the checking of usability, efficiency, and integrity of developed application serving its purpose

- ***Release and Maintenance*** is the final procedure of SDLC. This procedure witnesses the release of the actual software into the real world for practical use. In order to ensure the continuous efficiency of purpose served, developed application needs to be maintained and upgraded according to requirement

The development of the BMI/BMR calculation program-specific steps of SDLC has been followed for proper efficiency of the product designed.

3. Methodology

In this particular topic of discussion, the various methods of proper implementation of designed BMI/BMR checking program have been provided with specific detail to every single step involved. In the screenshot presented below a proper overview of the basic coded structure for BMI/BMR, calculation has been provided with the formulas implemented within. This approach ensures that the programming concept is on the right path. For the other options, the particular core java concepts like inheritance, polymorphism, and use of loops are to be defined with gradual and systematic development of processes.

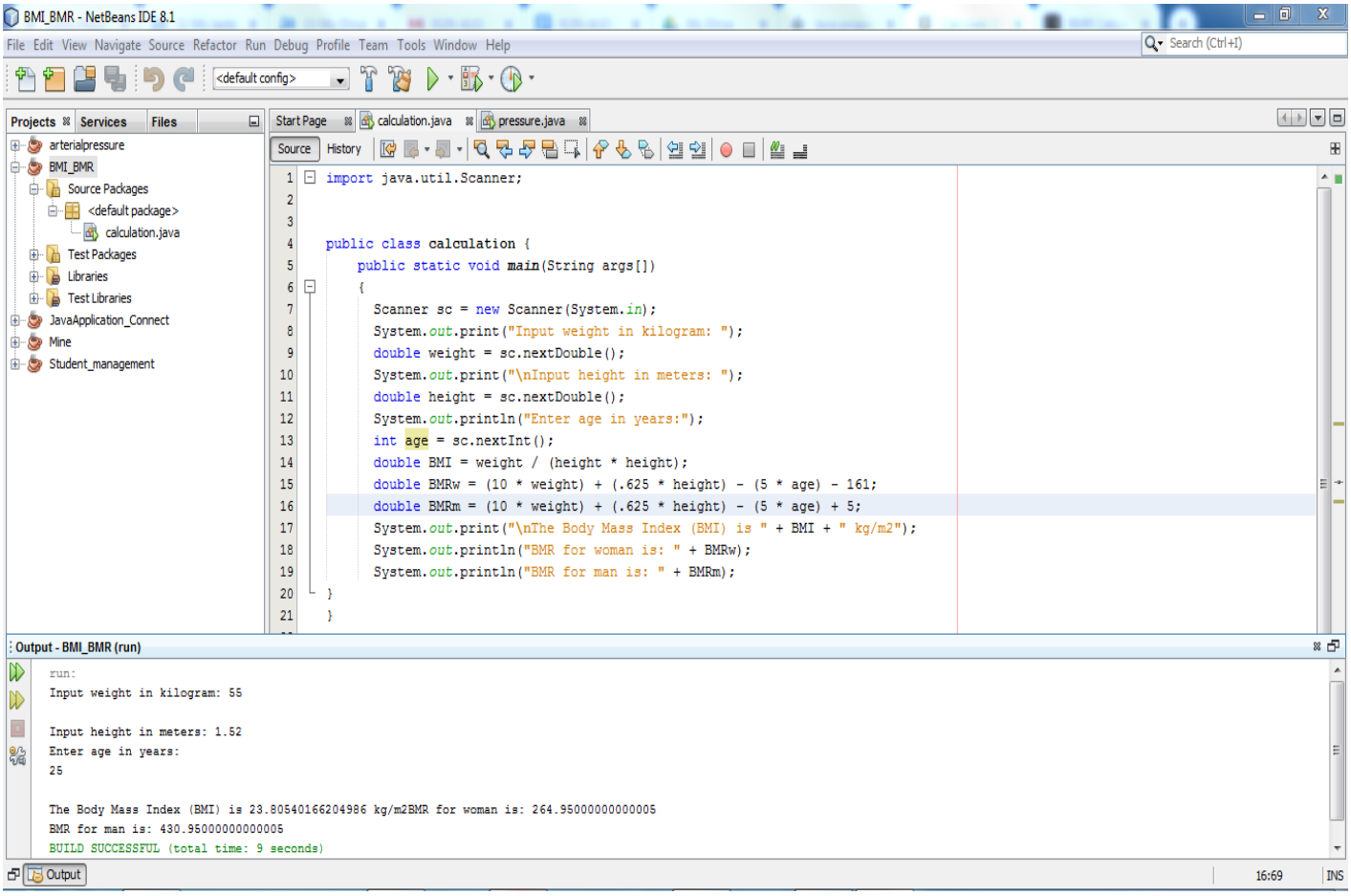

Figure 1: Basic BMI/BMR program implementation

Above presented screenshot provides the basic code structure for the implementation of formulas regarding Basal Metabolic Rate and Body Mass Index check for the user. The screenshot presented below provides specific coded structure for the menu-driven optional provided to users for the separate checking of BMI/BMR for humans and animals. In this particular menu structure, "switch case" method has been implemented for provision of various options as required by the developer to be presented to the concerned user. This particular programming will help in separately assigning databases for humans and animals.

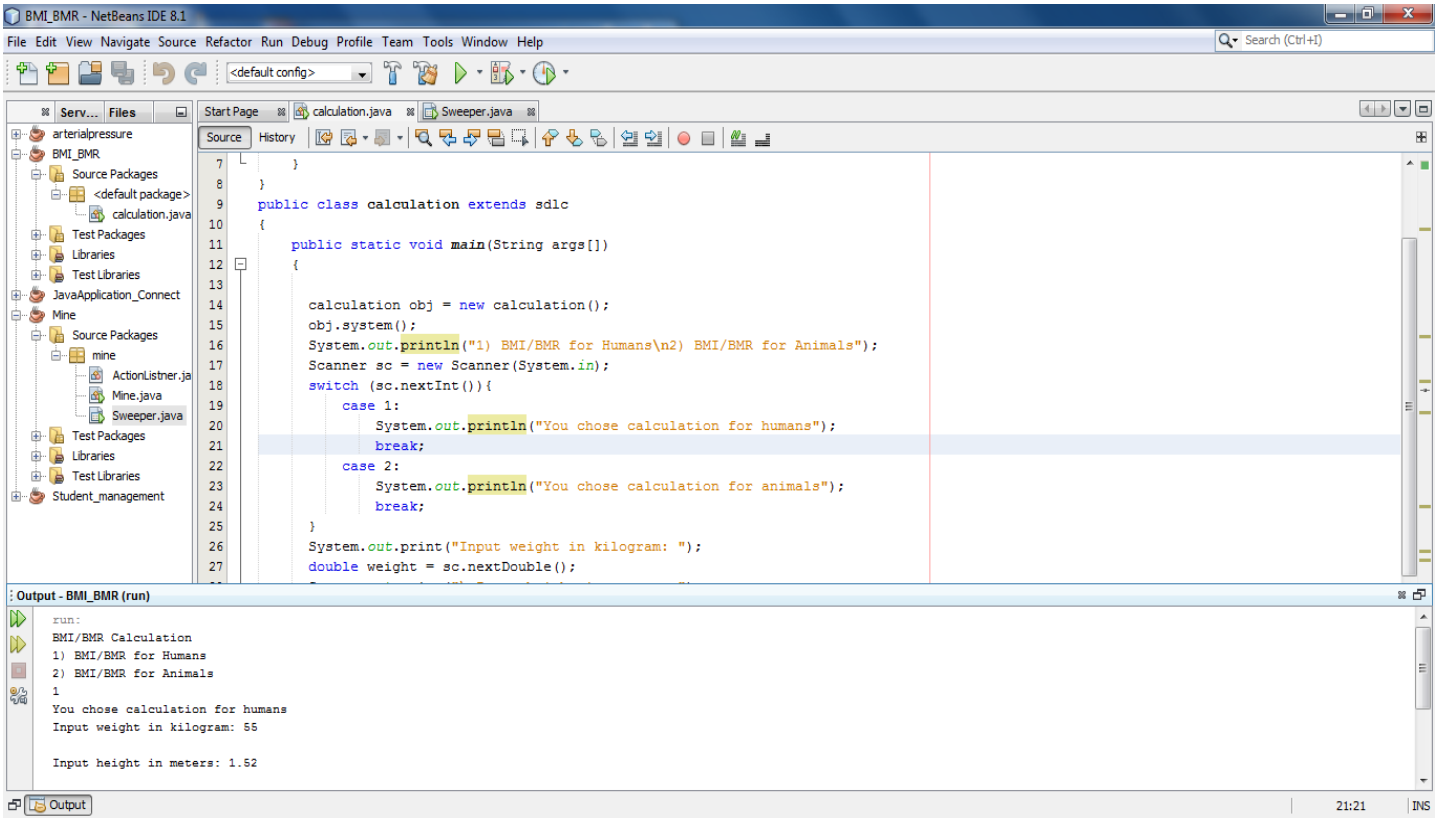

Figure 2: Choosing options between humans and animals

The provision of inheritance and polymorphism has been observed within particular coded structure according to requirement. This particular structure designed provides a proper overview of how classes can be extended, and objects can be derived from other classes according to specific requirement. This particular procedure also provides two separate functions for specific values taken not more than once where required. In order to check the Body Mass Index, the values of height and weight are required as input, while in case of Basal Metabolic Rate checking, age is also required to be initialized. This program takes the three inputs only once and presented the BMI first and then with the additionally provided age input and the previously provided height and weight input calculated the BMR. Core Java concepts of inheritance and polymorphism are thus served.

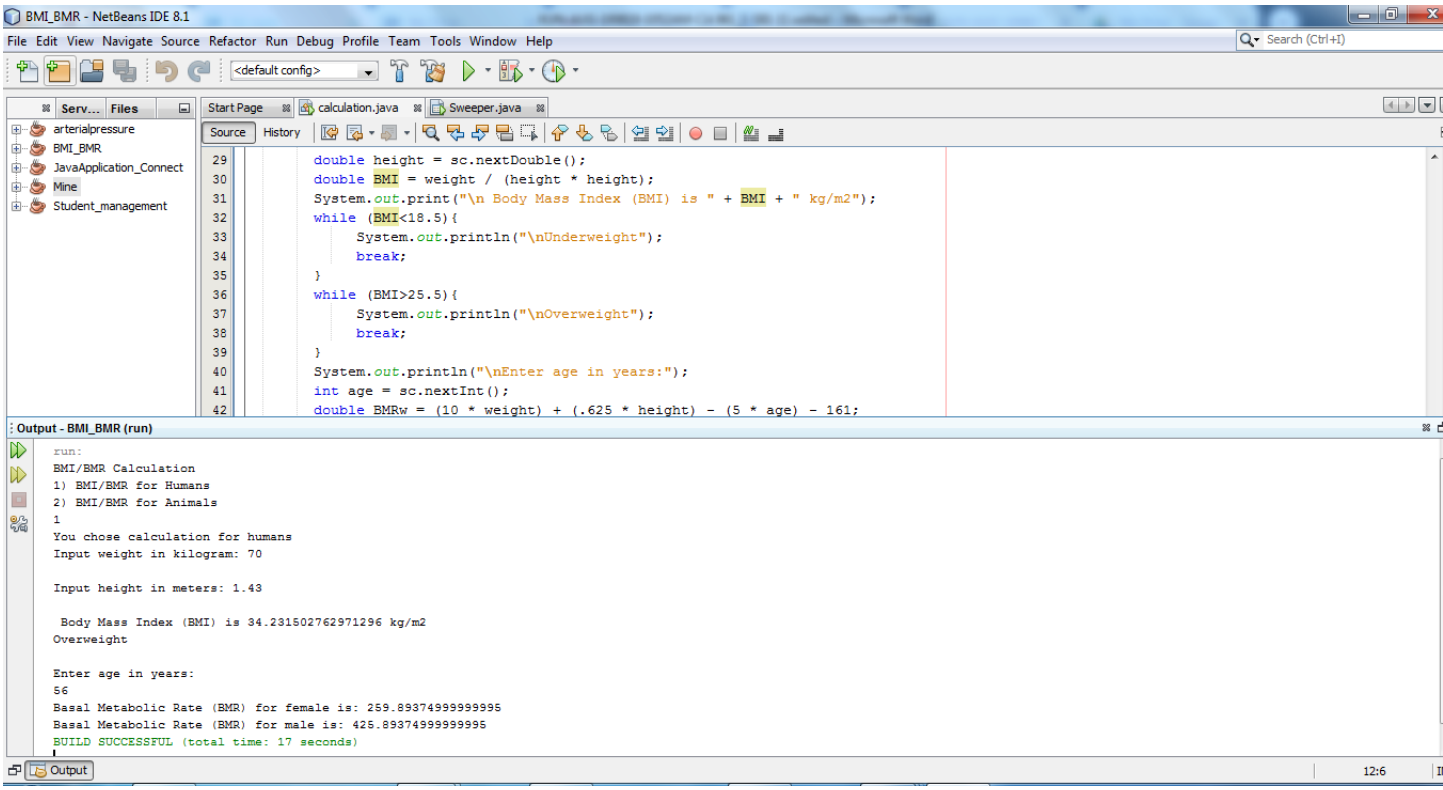

Figure 3: Use of loops

In this particular screenshot, the use of loops has been provided as a reference to the specific requirements. These loops have been initialized to consider the nature of the output of specific BMI/BMR calculations provided. Specific use of loops has been done to show the statuses ‘underweight', ‘overweight' and ‘normal'.

4. Requirement Analysis

In the application, the user can enter weight, height and age for calculating BMI and BMR. The application will accept height in meter and weight in kilogram. The program has been designed for calculating BMR and BMI of both human and animal. The user has to choose Human or Animal in starting of the program. In case of wrong selected option an error message will be displayed. As influenced by Dingsøyr *et al.* (2018, p. 502), the program will check the

input of user and switch to the function for finding the result for Animal or Human. If BMI will have value less than 18.5 then it will show result for Under weight and if BMI is greater than 25.5 then it will show result for overweight.

The validation of mass and body relationship required stastical analysis for the data. BSI and BMI include variation and it includes mean value for BSI. The appropriate relationship between the height and body mass is applied for scaling and it is rule for magnification of three dimensional bodies.

5. Concepts used in Java

In the program the concept of polymorphism has been used and it provide the characteristics for assigning differ usage and meaning with different context. It allows all entities like function, object and variable for having more than one form. As influenced by Hill *et al.* (2017, p. 8), the name of variable can allow having different program and form for determine the variable with the use during execution. The concept of polymorphism has been used in the program because it provides the way for distinguishing the form for handling every case and it can handled and recognized. The function provides the function that depends on the parameter. Another concept include in the program is inheritance. Inheritance includes the mechanism for acquiring the object with all behavior and properties form the parent object. The idea for inheritance can help in creating new class that is building on the existing class. Both have same name and during inheritance the existing call can also reuse the field and method of parent class. It can help in addition of field and methods for the current class. Extended keyword has been used for making new class and it can help in deriving the existing class. The terminology can help in inheriting the class.

Inheritance provides a pillar for the mechanism in java that allows class for inheriting the property of parent class. As influenced by Müller & Schaefer (2018, p. 18), the class with the feature of inheritance of class super class. Inheritance support concept for reusability and creating new class includes code that can be derived from new class. While loop statement helps in execution of the target statement and the provided condition is true. The program provides feature for facilitating the execution of instruction and function that can be used with condition is true. In While loop, the control statement for flow allows code for execution after reputation of the condition. The loop starts with checking the condition. In the loop the condition is checked, if the condition is true then the body statement can be executed further. If the statement is false then the loop will be executed. It is also called entry control loop. In this condition the valuation is true and the statement for the loop will be executed. The statement will include updated value and the variable will process for next iteration. If the statement for the condition will be false then the loop will be terminated that will mark the end for life cycle.

6. Implementation and Testing

The test plan can help in designing and provide understanding for testing and selection of type for testing. Every type of test type will have constraint and benefits. White box and black box testing can be used for the program. White box testing can run source file directly and every line of code can be tested with availability for measurement and inspection. It helps tester in resolving the failing tests with line by line step. The step for debugging also become easy for inspection and includes value for system and variables states. Black box testing is called functional testing. It works with the public interface for the service and application. Black test do not have access of interval for the artifact that are under the test. Black box testing is useful in

verification for the application and development of services and system. The tester and test engineer can conduct the test for analysis and entering the input for the resulting output. The input does not have direct output and testing can help in accessing some of the aspect of system.

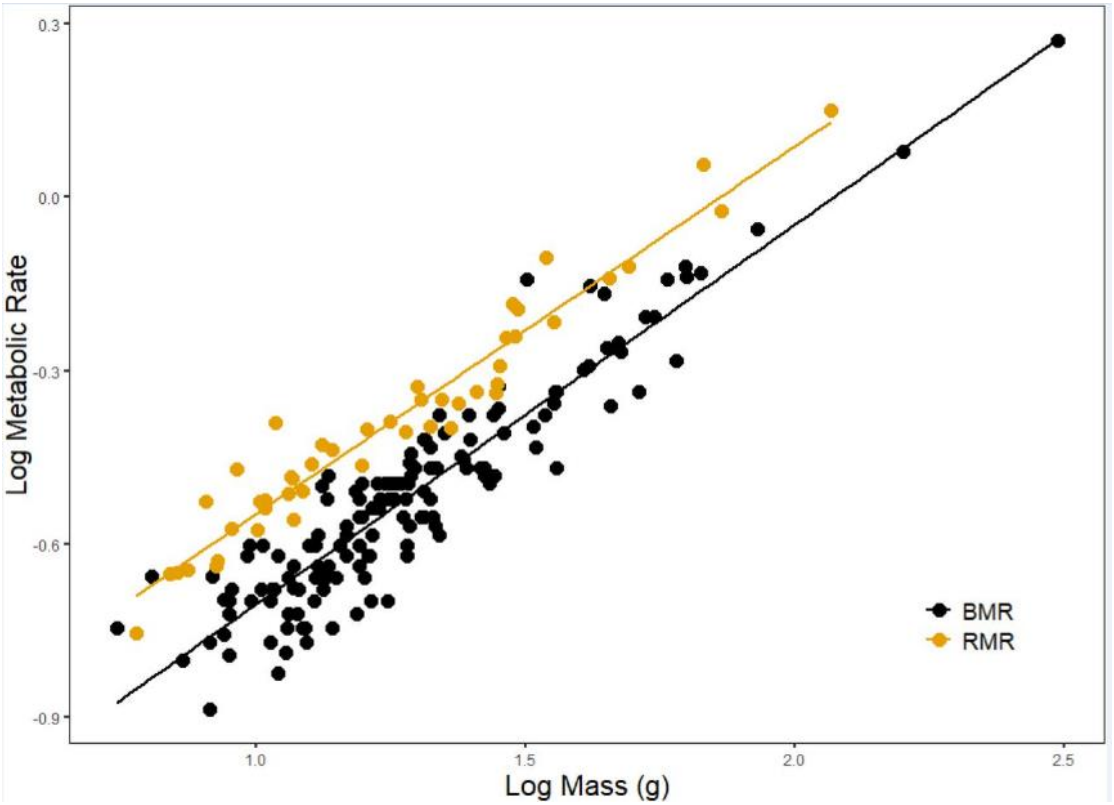

Figure 1: Database graph for human

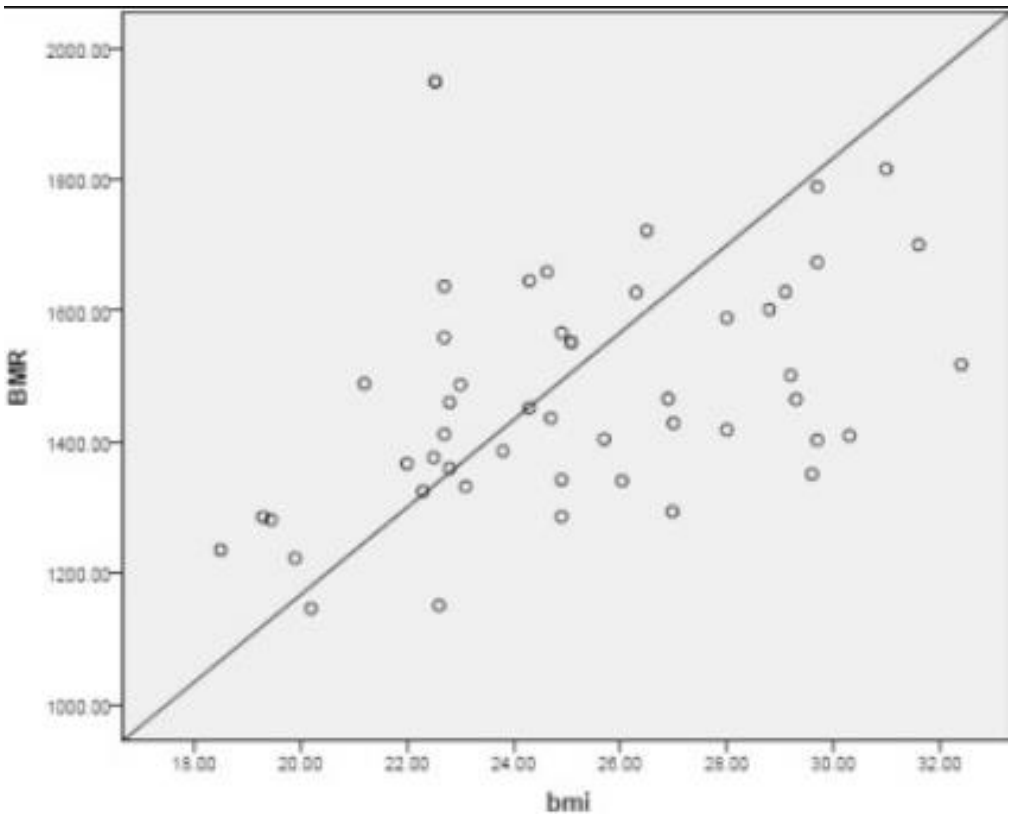

Figure 2: Database graph for animal

The application of software includes fabrication of software. Fabrication includes program designing and source code for testing every software unit. it is the series for technical test and this can help in representation of procedure of software and objects with the use of graphical model. Software fabrication is the documentation of source code that has been used for testing of structural unit. The software components are tested against the structural component and this can help in assuring correctness.

Conclusion

The report includes implementation of Java codes for finding BMI and BMR of animal and human by getting value for height, weight and age. The analysis of report provides with the approach for development and designing of the code. The program has used different

programming steps like inheritance, loop and polymorphism. The concepts have been used for development of program and obtaining desired results. The program can help in specifying the value that has been assigned by the user. After completion of program, testing methods like White and Black box testing has been used for ensuring that programis working properly and showing desired results.

References

Book

Müller, P., & Schaefer, I. (2018). *Principled Software Development*. Springer International Publishing.

Journals

Nazir, N., & Nazir, M. K. (2018). A Review of Security Issues in SDLC. American Scientific Research Journal for Engineering, Technology, and Sciences (ASRJETS), 46(1), 247-259. https://www.asrjetsjournal.org/index.php/American_Scientific_Journal/article/view/4241

Suryantara, I. G. N., & Andry, J. F. (2018). Development of Medical Record With Extreme Programming SDLC. IJNMT (International Journal of New Media Technology), 5(1), 47-53. http://ejournals.umn.ac.id/index.php/IJNMT/article/view/706

Telan, R. M., & Quioc, M. A. F. (2017). Forecasting the Inventory Needs for the City Schools Division of Mabalacat Using Moving Average and Exponential Smoothing. Universal Journal of Management, 5(9), 413-417. http://www.hrpub.org/download/20171030/UJM1-12109230.pdf

Tsui, R., Davis, D., & Sahlin, J. (2018, July). Digital Engineering Models of Complex Systems using Model-Based Systems Engineering (MBSE) from Enterprise Architecture (EA) to Systems of Systems (SoS) Architectures & Systems Development Life Cycle (SDLC). In INCOSE International Symposium (Vol. 28, No. 1, pp. 760-776). https://s3.amazonaws.com/academia.edu.documents/58477233/revised_IP_04-16-18_Paper49FinalSubmissionIS2018_manuscript.pdf?response-content-disposition=inline%3B%20filename%3DDigital_Engineering_Models_of_Complex_Sy.

pdf&X-Amz-Algorithm=AWS4-HMAC-SHA256&X-Amz-Credential=AKIAIWOWYYGZ2Y53UL3A%2F20190812%2Fus-east-1%2Fs3%2Faws4_request&X-Amz-Date=20190812T043929Z&X-Amz-Expires=3600&X-Amz-SignedHeaders=host&X-Amz-Signature=353c568ccfbffbe467da41159afda0abb284e516f64917d71111256d3c182f7b

Online Articles

Dingsøyr, T., Moe, N. B., Fægri, T. E., & Seim, E. A. (2018). Exploring software development at the very large-scale: a revelatory case study and research agenda for agile method adaptation. *Empirical Software Engineering*, *23*(1), 490-520. [Online] Retrieved on: 11 August 2019, Retrieved from: https://link.springer.com/article/10.1007/s10664-017-9524-2

Hill, A., Loh, P. R., Bharadwaj, R. B., Pons, P., Shang, J., Guinan, E., ... & Jelinsky, S. A. (2017). Stepwise distributed open innovation contests for software development: acceleration of genome-wide association analysis. *Giga Science*, *6*(5), gix009. [Online] Retrieved on: 11 August 2019, Retrieved from: https://academic.oup.com/gigascience/article/6/5/gix009/3057432

Šmite, D., Moe, N. B., Šāblis, A., & Wohlin, C. (2017). Software teams and their knowledge networks in large-scale software development. *Information and Software Technology*, *86*, 71-86. [Online] Retrieved on: 11 August 2019, Retrieved from: https://www.sciencedirect.com/science/article/pii/S0950584917300435