

#### VISVESVARAYA TECHNOLOGICAL UNIVERSITY BELAGAVI, KARNATAKA

###### A Project Work Phase-1 Report on

“Software Cost Estimation using Back Propagation Algorithm”

Submitted in the partial fulfillment for the award of

##### BACHELOR OF ENGINEERING

**in**

##### INFORMATION SCIENCE AND ENGINEERING

### By

**Sudeep Rangan D R Samuel Sampath Kumar**

**1BY19IS162 1BY19IS144**

**Anamika Sharma Apurv Jha**

**1BY18IS129 1BY18IS028**

### Under the guidance of

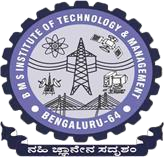
**Mr. Ravi Kumar B N**

Associate Professor Dept. of ISE BMSIT&M

##### 

##### 2022-2023

#### BMS INSTITUTE OF TECHNOLOGY & MANAGEMENT



YELAHANKA, BENGALURU-560064

DEPARTMENT OF INFORMATION SCIENCE & ENGINEERING

This is to certify that the Project work Phase-1(18CSP77) entitled **“Software Cost Estimation Using Back Propagation”** is a bonafide work carried out **by Miss. Anamika Sharma (1BY18IS129), Mr. Apurv Jha(1BY18IS028), Mr. Sudeep Rangan D R (1BY19IS162), Mr. Samuel Sampath Kumar(1BY19IS144)** in partial fulfillment for the award of Bachelor of **Engineering Degree in Information Science and Engineering** of the Visvesvaraya Technological University, Belagavi during the year 2022-23. It is certified that all corrections/suggestions indicated for Internal Assessment have been incorporated in this report. The project report has been approved as it satisfies the academic requirements with respect to project work for the B.E Degree.

Signature of the Guide Signature of the Coordinator

Mr. Ravi Kumar B N Dr. Basavaraj G N

Signature of the HOD Signature of the Principal

Dr. Mohan Babu G N

**ABSTRACT**

Software cost estimation is one of the most challenging tasks in software engineering. Over the past years the estimators have used parametric cost estimation models to establish software cost, however the challenges to accurate cost estimation keep evolving with the advancing technology. This project uses the Back-Propagation a neural networks for software cost estimation. A model based on Neural Network has been proposed that takes the KLOC of the project as input, uses COCOMO model parameters and gives cost as output. Artificial Neural Network represents a complex set of relationships between the effort and the cost drivers and is a potential tool for estimation. The proposed model estimates the software cost and helps in project manager to provide fast and realistic estimates for the project effort and development time that in turn gives cost

# ACKNOWLEDGEMENT

We are happy to write a project report after completing it successfully. This project would not have been possible without the guidance, assistance and suggestions of many individuals. We would like to express our deep sense of gratitude to each and every one who has helped us to make this project a success.

We heartily thank **Dr. Mohan Babu G. N, Principal, BMS Institute of Technology & Management** for his constant encouragement and inspiration in taking up this project.

We heartily thank **Head of the Department, Information Science and Engineering, BMS Institute of Technology& Management** for constant encouragement and inspiration in taking up this project.

We gratefully thank our project guide**, Mr. Ravi Kumar B N, Associate Professor Dept. of Information Science and Engineering,** for his encouragement and advice throughout the course of the project work.

We heartily thank project coordinator **Dr. Basavaj G N, Associate Professor, Dept. of Information science and Engineering,** for their constant follow-up and advice throughout the project work

Special thanks to all the staff members of the Information Science and Engineering Department for their help and kind co-operation.

Lastly, we thank our parents and friends for their encouragement and support given to us in order to finish this project work.

By,

**Samuel Sampath Kumar**

**Sudeep Rangan D R Anamika Sharma**

**Apurv Jha**

|  |  |  |
| --- | --- | --- |
|  | **BMS INSTITUTE OF TECHNOLOGY & MANAGEMENT**  **YELAHANKA, BANGALORE-64**  **DEPARTMENT OF INFORMATION SCIENCE AND ENGINEERING** | C:\Users\SKP\Pictures\DeptLogo.png |

**Declaration**

We, hereby declare that the project phase-1(18CSP77) titled “SMART SHELVES IN A RETAIL STORE” is a record of original project phase-1 work undertaken for partial fulfilment of Bachelor of Engineering in Information Science and Engineering of the Visvesvaraya Technological University, Belagavi during the year 2022-23.We have completed this project phase-1 work under the guidance of  **Mr. Ravi Kumar B N. Associate Professor**.

I also declare that this project phase-1 report has not been submitted for the award of any degree, diploma, associate ship, fellowship or other titles anywhere else.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Student Photo :-** |  |  |  |  |
| USN:  Name: | 1BY18IS129  Anamika Sharma | 1BY19IS144  Samuel Sampath Kumar | 1BY18IS028  Apurv Jha | 1BY19IS162  Sudeep Rangan D R |
| Signature | : |  |  |  |

**CONTENTS**

ABSTRACT i

[ACKNOWLEDGEMENT ii](#_bookmark0)

[DECLARATION iii](#_bookmark1)

[LIST OF FIGURES v](#_bookmark2)

|  |  |
| --- | --- |
| 1. **Introduction**    1. Introduction    2. Motivation    3. Problem Statement    4. Proposed System    5. Objectives | **1**  1  2  3  3  4 |
| 1. **Literature Survey**    1. Critical Analysis of the literature    2. A Summary Table    3. Implication and Conclusion    4. Existing System | **5**  5  13  18  19 |
| 1. **System Requirement Specification**    1. Functional Requirements    2. Non-Functional Requirements    3. Hardware Requirements    4. Software Requirements | **21**  21  21  22  23 |
| 1. **Design And Analysis**    1. Data Collection & Survey    2. System Design | **24**  24  24 |
| 1. **Methodology**    1. Steps 2. **Conclusion**    1. Conclusion | **25**  25  **27** |

# LIST OF FIGURES

|  |  |  |
| --- | --- | --- |
| **Figure No.** | **Figure Name** | **Page No.** |
| 4.1  5.1 | Smart Shelf Technology Shapes Retailing  Integrating the modules | 24  26 |

## CHAPTER 1

**INTRODUCTION**

##### Introduction

The software cost estimation for Information Systems is process used by an organization in order to forecast the cost for the development of a software project. Effort estimation is the process of forecasting how much effort is required to develop or maintain a software application. This effort is traditionally measured in the hours worked by a person, or the money needed to pay for this work. Effort estimation is used to help draft project plans and budgets in the early stages of the software development life cycle. This practice enables a project manager or product owner to accurately predict costs and allocate resources accordingly. Software cost estimation is one of the most challenging tasks in software engineering. Over the past years the estimators have used parametric cost estimation models to establish software cost, however the challenges to accurate cost estimation keep evolving with the advancing. This project uses Back-Propagation neural networks for software cost estimation. A model based on Neural Network has been proposed that takes the KLOC of the project as input, uses COCOMO model parameters and gives cost as output. Artificial Neural Network represents a complex set of relationships between the effort and the cost drivers and is a potential tool for estimation. The proposed model estimates the software cost and helps project manager to provide fast and realistic estimates for the project effort and development time that in turn gives software cost. This dataset contains numerical values as well as categorical data. Within this dataset there are a high percentage of missing values. Due to this, the data mining techniques are used for preprocessing the information. The size estimation is the measuring of the project size, usually in lines of code or equivalent. Since software is a product without physical presence and the main cost is the design and development of the product, the cost is dominated by the cost of the human resources, measuring this effort in man-months

##### Motivation

Speedy and accurate project cost estimation is critical in order to ensure customer satisfaction and repeat business. However, it remains one of the most arduous tasks in software engineering, especially when the projects dealt with are complex, extensive, and in the conceptual stage. The advancements in ANN technologies are transforming the traditional methodology of software cost estimation into a flexible and intelligent approach. A compelling use case is an intelligent solution to identify and eliminate duplicates, analyze and pin-point ambiguity, and process RFPs in varied formats with minimal human intervention, for efficient and expedited project cost estimation. An intelligent system leveraging ANN can be used to overcome the human errors and arrive at meaningful estimates within limited time frames. The solution empowers project managers to automatically analyze requirements, weigh dependencies, and generate detailed reports, resulting in expedited software cost estimation.

##### Problem Statement

Software cost estimation is the set of procedures and techniques with a set of inputs that an organization uses to achieve a software cost estimate in terms of effort, manpower, duration etc. Generally, inaccurate estimates are because of:

● More Hidden Layers: Usually assumed that more layers means faster processing and results but in the case of back propagation algorithm 2 layers are proven to be more effective than 3,4,5 hidden layers.

● Increasing Complexities: As there are many factors involved in creating a software it becomes highly complex in predicting the accurate cost of the software therefore the applications regarding software cost estimation also becomes complex. Problem with requirements, System Size, Maintenance Issues, Software Process and Process Maturity

● Time Delays: Using other machine learning models and algorithms the expected accuracy is not Reached and hence as these are also complex and comprise more layers, time delays are inevitable.

##### Proposed System

The existing system is too bulky and complex with noticeably more accuracy for costlier and slower hence the proposed system is less complex, easy to understand and use. This system is to be developed with the help of the back propagation algorithm of Neural Networks. To provide a practical and effective approach for preparing data and building models is applied and presented based on the ISBSG dataset, which provides the most reliable source of a large volume of recent software projects from multiple industries (International Software Benchmarking Standards Group) with ANN predictive algorithms. Back Propagation (BP) is a method for training multilayer feed forward networks (Fadalla et al., 2001). It works by training the output layer and then propagating the error calculated for these output neurons, back though the weights of the net, to train the neurons in the inner (hidden) layers.

##### Objectives

* Designing a Software cost estimation model for Object Oriented Systems.
* Applying ANN techniques for the basic model to improve the accuracy.
* Comparing with existing techniques and to prove the propose Software Effort Estimation model is more Efficient.
* The purpose of this project is to narrow the gap between up-to-date research results and implementations within organizations’ by proposing effective and practical machine learning deployment and maintenance approaches by utilization of research findings and industry best practices.
* We want to achieve this by applying ISBSG dataset, smart data preparation, machine learning algorithms (COCOMO and Back Propagation) and cross validation and provide a more accurate model than the existing mode

**CHAPTER 3**

**LITERATURE SURVEY**

* 1. Critical Analysis of the Literature Survey
* In this Paper [1], Global Software Development (GSD) projects comprise several critical cost drivers that affect the overall project cost and budget overhead. Thus, there is a need to amplify the existing model in GSD context to reduce the risks associated with cost overhead.Motivated by this, the current work aims at amplifying the existing algorithmic model with GSD cost drivers to get efficient estimates in the context of GSD.

* In this Paper [2], There are numerous tools and techniques have been developed for estimating the software cost. But all these techniques are best suitable for the traditional development methodology. From the past two decades, the agile methodology has been com for software development. It gives the appropriate cost estimation for agile development.

* In this Paper [3], MultiLayer Perceptron (MLPNN) and its ensembles are explored in order to improve the performance of software effort estimation process. Firstly, MLPNN, Ridge-MLPNN, Lasso-MLPNN, Bagging-MLPNN, and AdaBoost-MLPNN models are developed and, then, the performance of these models are compared on the basis of R2 score to find the best model fitting this dataset. Results obtained from the study demonstrate that the R2 score of AdaBoost-MLPNN is 82.213%, which is highest among all the models.

* In this Paper [4], NAYA application developed by researchers has guided developers to write supporting data on the calculation components that have been provided. This application was developed with an iterative development model for implementation because application development requires feedback from the user as an improvement of specific functions by the system.

* In this Paper[5], we are using a new meta-heuristic algorithm inspired by the strawberry plant for optimization of COCOMO effort estimation method. NASA 93 data set is used in the proposed approach. The Magnitude of Relative Error (MRE) and Mean Magnitude of Relative Error (MMRE) is evaluated. Experimental results of the proposed method with the COCOMO model shows a decline in MMRE to 23.8.

* In this Paper[6], Four different machine learning techniques have been compared for the estimation of effort required to The proposed methodology used in this paper consists of four machine learning models namely, LR, SVM, KNN, and MLPNN

* In this Paper[7], This paper will present the use of two techniques i.e., expert judgment and Constructive Cost Model (COCOMO) II to enhance the predictability of COCOMO II for the undergraduate projects of universities. It calculates the size through function point, use case points or object points metrics and converts them in kilo line of code (KLOC). The scheduled time for these projects is usually one year from its inception till deployment. Effort estimation of these projects is conducted through the use of tool Estimator.

* In this Paper[8], study mainly focuses on the optimization of the Constructive Cost Model (COCOMO)-II parameters by implementing Flower Pollination Algorithm (FPA) on standard datasets taken from Turkish Industries software project. The comparison graph of Manhattan distance (MD) and mean magnitude of relative errors (MMRE) indicate that cost estimation provided by the proposed FPA model is better than the original COCOMO-II model and Bat algorithms.

* In this Paper[9], DOLPHIN BAT ALGORITHM (DOLBAT) PROPOSED TO ESTIMATE SOFTWARE EFFORT In this work has been using dolphin algorithm and Hybridized with bat algorithm to get better coefficients value to predict effort of software, in the initial phase in dolphin algorithm it configure three matrices randomly.

* In this Paper[10], They have used Artificial Neural Networks with Cuckoo Optimization Algorithm to present as a new model for prediction of Software Cost Estimation. We verified this technique on the International Software Benchmarking Standards Group (ISBSG) licensed dataset, which contains 1531 projects.

* In this Paper[11], The advantage of this method is that this method takes valuable experience of expert, which can suitable for more complex scenario and more general topic it depends too much on the expert, which lacks standardization and can be subjective. Challenges raised in Function Point (FP) analysis have been described in many research and industry practice, there are two most obvious barriers to conduct FP analysis, and the first is the rule to define the definition of different type of function point.

* In this Paper[12], This paper proposed a method called data smoothing to mitigate the problem of outliers without reducing the number of data points in a data set. The proposed method follows the assumption of Analogy-Based Estimation (ABE) such that “projects with similar features require similar development efforts,” and changes the effort values in a data set so as to satisfy this assumption.

* In this Paper[13], Analysis and designing using analogy based estimation method, that are Building a new dataset using data collection of previous software projects, Proposing the K-Nearest Neighbor (KNN) with mean and Inverse Distance Weight (IDW) interpolation technique for the effort calculation method, and Designing the cost allocation model based on the regulations of goods and service procurement in government institutions in Indonesia.

* In this Paper[14], In the United States Department of Defense (US DoD), it is necessary and most critical to estimate software development cost in early lifecycle phases when limited data is available. The model accuracy improves after peak staff and super domain are incrementally added to the model.

* In this Paper[15], As can be drawn from global, regional and local experiences with software project estimation, there is no one overriding factor that causes project failure or overrun, a number of factors are involved in any particular project failure, some of which interact with each other.

* In this Paper[16], Software Project Management (SPM) included a series of interrelated processes which were to be carried out in a specific manner with the development process. Recent research and practical outcomes of different software developments have proved that prediction of cost and effort estimation with a high rate of accuracy increases the chance of successful quality product.

* In this Paper[17], Analysis and designing using analogy based estimation method, that are Building a new dataset using data collection of previous software projects, Proposing the K-Nearest Neighbor (KNN) with mean and Inverse Distance Weight (IDW) interpolation technique for the effort calculation method, and Designing the cost allocation model based on the regulations of goods and service procurement in government institutions in Indonesia.

* In this Paper[18], The main goal of this study is to evaluate software effort estimation models that are built based on machine learning approaches. This paper presents various software effort estimation models using four machine learning algorithm.

* In this Paper[19], The traditional COCOMO need some extended feature for accurate calculation of efforts in these new approaches. Bohem evolve the new version of COCOMO in which he considered some more factors involved in effort estimation. Detailed COCOMO: Detailed COCOMO focuses on phase wise effort calculation for different phases in SDLC.

* In this Paper[20], In order to develop a system which improve the metrics for automotive industry, the square's devil above should be taken in account. Automotive software is safety-critical so product quality is high priority.The iterations during the development cycle will involve many change requests: some to add or edit requirements and features, and others to correct defects which originated in a different phase of the cycle.

##### A Summary Table

Software project development has become extremely complicated, and the necessary competence in this industry is high, which requires the skills of highly qualified people. In past decades, to complete a project and deliver it to the customer on time, schedule, and budget, project managers had to estimate the cost of the software product, effort, and project duration or defect density. Developing a reliable parametric cost model at the conceptual stage of the project is crucial for project managers and decision makers. Existing methods, such as probabilistic and statistical algorithms have been developed for project cost prediction. However, these methods are unable to produce accurate results for conceptual cost prediction due to small and unstable data samples. Artificial intelligence (AI) and machine learning (ML) algorithms include numerous models and algorithms for supervised regression applications.

The individual analysis of the various proposed systems has given us a lot of insights to frame the design our proposed model. We also came across various performance evaluation metrics to compare the models.

We came across different technologies such as Artificial neural network[ANN],deep learning and meta-heuristic algorithms used to improve the efficiency of the cost prediction model. Some models have been trained and validated over a large number of datasets but performance evaluation is done using a smaller number of metrics.

Some models were not compared with classic models and few systems have been trained and validated against small dataset. Considering all these factors we have designed our model by considering various evaluation metrics against classic models like COCOMO model.

* 1. **A Summary table**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **S.No** | **Author** | **Approach** | **Approach** | **Pros** | **Cons** |
| 1. | JUNAID ALI KHAN, SAIF UR REHMAN KHAN, TAMIM AHMED KHAN, AND INAYAT UR REHMAN KHAN  IEEE - June 2021 | Muhammad Ehsan Rana  Junie Halim  Kamalanathan Shanmugam  IEEE,2020 | The researchers have proposed the implementation of iBeacon based smart shelves for retail stores based on IoT devices and other infrastructural components using a systematic approach. | iBeacon has a internal clock which can use to track of the expiry date | Need for automatic  product recognition |
| 2. | Design of Smart Unstaffed Retail Shop Based on IoTand Artifical Intelligence | Jianqiang Xu Junzhong zou IEEE,2021 | To analyze multiple target features of commodities, the SSD (300 × 300)  algorithm is employed. | Merger of IoT and AI successfully achieved | Need for better algorithm for recognition rate andefficiency |
| 3. | A Bluetooth Location- basedIndoor Positioning System for Asset Trackingin Warehouse | C.K.M. Lee  C. M. Ip IEEE,2021 | Introduction of Kalman-LULU filter, in signal processing stage, helps increase the accuracy ofthe system and reduce noise in fast pace. | Kalman-LULU filter has better processing system than other filters available in the market | Need sensors for better evaluation of distance to achieve higher precision |
| 4. | A  Location- Based Smart Shopping System withIoT Technology | Javed Rezazadeh  Kumbesan Sandrasegaran  IEEE,2020 | Based on four components include location of everything component, data collection component, data filtering/analysing component data  mining component | GPS and IoT merger achieved | Need of more precisecomponent is needed |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 5. | A survey of Internet of Things: Future Vision, Architecture, Challenges and Services | Gaurav Tripathi  Dhananjay Singh  IEEE,2021 | This architecture introduces the use of Smart Semantic framework to encapsulate the processed informationfrom sensor networks | Proper implementation ofSmart Semantic framework | The present  architecture has the scope to improve a lot on the semantic and security front |
| 6. | Smart Shelf Using Internet of Things | S.Subramanian V.Lalitha IEEE,2021 | The system provides easy monitoring of stockinventory by using load sensors. | Implementation ofload sensors | Calibrating  ultrason icsensors is expensive |
| 7. | Context-Aware Computing, Learning, and Big Data in Internet of Things: A Survey | Ipek Baz Erdem Yoruk IEEE,2018 | The researchers proposed a hybrid context-aware product recognition system thatclassifies products. | Merger of IoT andBig Data | Extend the model to 2D with spatial product configurations  on  shelves  includi ng horizontal and vertical adjacencies. |
| 8. | Internet of Things: A Survey on Enabling Technologies, Protocols, and Applications | Ala Al-Fuqaha  Mohsen Guizani  IEEE,2019 | The researchers presented the need for new “smart” autonomic management, data aggregation, and protocol adaptation services to achieve betterhorizontal integration among IoT service.  Finally, detailed application-use cases were presented to illustrate typical protocolintegration  scenarios to deliver desired IoT services. | Horizontal integrationof IoT devices | Better intergration of IoT devices will lead to better result |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 9. | DiffNet: A Learning to Compare Deep Network for Product Recognition | Bin Hu Qiang Zhou IEEE,2020 | The researchers have proposed a system basedon the image set of 1846images | Big Dataset is used | Bigger dataset willproduce  better result. |
| 10. | Big Data Analysis-based Secure Cluster Management for Optimized Control Plane in Software- Defined Networks | Jun Wu  Mianxiong Dong  IEEE,2020 | This paper proposed a big data analysis-based secure cluster management architecturefor optimized control plane. A secure authentication scheme was proposed to ensure the legality of the data sources. | Authentication isdone | A distributed security data storage scheme for the SDN controller cluster will be proposed |
| 11. | Knowledge Tradable in Edge-AI Enabled IoT: A Consortium Blockchain Based Efficient and Incentive Approach | Xi Lin Jun Wu  IEEE,2021 | In this paper, to break islands of knowledge and make knowledge tradable in edge-AI enabled IoT, they proposed a P2P knowledge market for knowledge paid sharing. | Merger of IoT andBlockchain is achieved | Mass scale implementation has tobe achieved. |
| 12. | Smart Shelfie – Internet of Shelves | Rakesh Satapathy  Srikanth Prahlad  IEEE,2018 | Researchers have enabled shelves to take its own selfie using a simple camera and then the image processing algorithms take over to do required analytics andraise alerts with the rightperson | Automatic shelfies has been achieved forproduct recognition | Reduction of cost is to be made |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 13. | Intelligent-loT Based Digital Racking System For Retail Shops | Shiva M C  Prakash Ramachandran  IEEE,2020 | Objective of detection,recognition and findinginstances of an information from a model has been achieved. | Single model for detection, recognitionis achieved | Implementation  on large-scale is to be achieved |
| 14. | Context-aware hybrid classification system for fine- grained retail product recognition | Ipek Baz Erdem Yoruk IEEE,2019 | The fundamental goal ofthis paper is using contextual relationships in retail shelves to improve the classification accuracy by executing a  context- aware approach. | Context-aware approach is achieved. | To extend our model to 2D with spatial product configurations  on  shelves  includi ng horizontal and vertical adjacencies. |
| 15. | Smart Shelve Management System using IOT-A Novel Approach | Nilesh Korde  Abhijeet Thakare  IEEE,2019 | The system with the helpof weight and level sensor helps the user to identify required food items in a specific shelve(s) of a specific rack system | Level sensors is usedto keep track of the products | More precise sensors can be introduced for faster better results |
| 16. | Intelligent IoT Shelf Design and Development | László Somai László Molnár IEEE,2018 | Weight Sensors are usedto identify whether thereare products are in the shelf or not | Weight sensors is used to keep track ofthe products | Need to improve  operational effiencies |
| 17. | Vision Based Intelligent Shelf- Management System | Priyanwada H.A.M  Nicoleta STROIA  IEEE,2021 | An original low cost system has been implemented as a solution for large setups of shelves on which items with RFID tags arestored | RFID tags gives morecoverage | Better precision can be achieved with the sensors |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 18. | A Low Cost Approach to Large Smart Shelf Setups | Florin Hrebenciuc  Daniel Moga IEEE,2020 | The proposed setup is a fully wireless one. It contains a standard reader, electromechanical positioning actuators andwireless communication and control hardware  offering power from integrated batteries. | Low cost | RFID(Radio Frequency Identification) tags can be improved |
| 19. | Digitalization at the Point-of- Sale in Grocery Retail - State of the Art of Smart Shelf Technology and Application Scenarios | Marike Kellermayr- Scheucher  Laura Hörandner  IEEE,2018 | Use of two main technology:  Image Recognition Systems  Sensor Based Systems | IMR are used for product identification | There is also a need for research to shed more light on the user side. On the one hand, this involves retailers, their acceptance, barriers, needs and future plans. |
| 20. | Study of smart inventory management system based on the internet of things (Iot) | Souvik Paul  Atrayee Chatterjee  IEEE,2019 | This paper presents a new type of intelligent Inventory Management System based on the IoTand explains the principles and structure of it | Large Scale implementation isseen | It can be used in several areas in different applications and many enhancements can be done so that it can be made available to all the sectors. |

##### Implication and Conclusion

Based on the detailed literature survey we infer that we will train and validated our model against ISBSG dataset. The cost prediction model will be COCOMO. The estimated size of the software product in thousands of Delivered Source Instructions (KDSI) adjusted for code reuse. The project development mode given as a constant value B (also called the scaling factor). 15 cost drivers. The project development mode depends on one of the three categories of software development modes: organic, semidetached, and embedded. Each rating has a corresponding real number (effort multiplier), based upon the factor and the degree to which the factor can influence productivity. This was the brief on implication and conclusion.

##### Existing System

All the methodologies of project management make management of plan and costs in any type of project and in the projects of software. The chosen system to make the estimations has to have the confidence of the project management and to allow to adapt again to the changing necessities of the software. Nowadays, there are several models available of cost estimation like COCOMO model, Checkpoint, ESTIMACS, SLIM, Knowledge Plan, etc.).

Among all of them, COCOMO model is one of most open and well-documented cost estimation models. At present, most of real-world use of cost models is bottom-up rather than top-down. The historical data summary in the end of the project is essential to update the data base of projects and so that the system can fit its parameters to the changing conditions of software.

There are various hybrid systems proposed which is compared and evaluated against basic models. Few of the research teams have conducted a comparison by investigating 20 different AI techniques but considering only very few performance metrics. Some models are trained against randomly selected projects from a dataset consisting few projects. Some hybrid models are not compared with classic COCOMO models to prove hoe reliable they are for real world implementation. Also many models have been trained and validated against standard dataset by ignoring how heterogeneous the data is and proper feature selection is not done.

**CHAPTER 3**

**SYSTEM REQUIREMENT ANALYSIS**

##### Functional Requirements

• Usability: The model should be able to predict effort and cost estimate of all categories of systems and projects

• Reliability: The effort estimated by the model must be accurate and reliable which in turn helps the project managers to proceed.

• Performance: The system should be consistently responsive and with a good performance.

• Robust: To handle complex situations and provide results

• Efficiency: The system should work efficiently while gathering data

##### Non-Functional Requirements

* + - Security
      * System needs to control the user access and session
      * It needs to store the data in a secure location and stored in a secure format
      * It requires a secure communication channel for the data.
    - Concurrency and Capacity
      * System should be able to handle multiple computations executing simultaneously and potentially interacting with each other.
    - Performance
      * Performance is generally perceived as a time expectation. This is one of the most important considerations especially when the project is in the architecture phase.
    - Reliability
      * It is necessary to ensure and notify about the system transactions and processing as simple as keep a system log increases the time and effort to get it done from the very beginning. Data should be transferred in a reliable way and using trustful protocols.
    - Maintainability
      * Well-done system is meant to be up and running for long time. Therefore, it will regularly need preventive and corrective maintenance. Maintenance might signify scalability to grow and improve the system features and functionalities.
    - Usability
      * End user satisfaction and acceptance is one of the key pillars that support a project success. Considering the user experience requirements from the project conception is a win bet, and it will especially save a lot of time at the project release, as the user will not ask for changes or even worst misunderstandings.
    - Documentation
      * All projects require a minimum of documentation at different levels. In many cases the users might even need training on it, so keeping good documentation practices and standards will do this task spread along the project development; but as well this must be establish since the project planning to include this task in the list.

##### Hardware Requirements

• Processor: Minimum 1 GHz; Recommended 2GHz or more

• Ethernet connection (LAN) OR a wireless adapter (Wi-Fi)

• Hard Drive: Minimum 32 GB; Recommended 64 GB or more

• Memory (RAM): Minimum 1 GB; Recommended 4 GB or above

##### Software Requirements

* + Python (3.7.4)
  + IDE (Jupyter)

**DESIGN AND ANALYSIS**

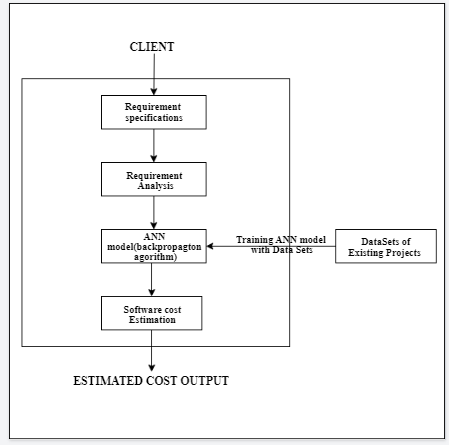
****

Fig: 4.1 Design o**f** the cost estimation model

##### Steps Involved in estimating the software cost through this project:

##### ● Step 1: Getting the requirement specification from the client

##### ● Step 2: Analyzing the processing the requirement specifications given by the client

##### ● Step 3: Feeding it the created software which is trained with datasets (These data sets contain the already created software details which include the overall cost of that software)

##### ● Step 4: The software processes the given data with the back propagation algorithm and the cost is estimated and given as output

**METHODOLOGY**

The overall design of the project revolves around an shelf store which can be used to perform multiple functionalities. The process of building our shelf has been divided into the following steps.

* + 1. Building basic IoT setup to collect data that aids retailers.
    2. Building a dashboard to collect data from individual shelf area and visualizing them
    3. Building a automated systems.
    4. Converting our setups into modules
    5. Integrate the modules to work together

Step 1: Building a basic IoT setup

The IoT setup comprises various swappable components. Some of the functionalities include: Product sensing using iBeacon

Step 2: Building a dashboard

A dashboard will be used to present the information being collected from the sensors as mentionedin the previous step. This will then be visualized so that retailers can understand and comprehend figures easily as less knowledge is required. Going forward, a custom dashboard will be built so as to integrate and streamline the various processes involved.

Step 3: Building a App

We build a application.The chassis will be based on the retail store that, however it will be remotely controlled using IoT components.

Step 4: Building modules that work with the application

We build modules that we can attach to the application

* We build the retail components and put it together into one module
* We take the other IoT based setups from step 1 and convert them into modules as well

*Software Cost Estimation using Back Propagation Algorithm*

Step 5: Integrating the modules and Finishing touches

There are some other considerations that come into picture here. For example, we cannot just integrate. We take such considerations into account here to solve them. For example, the above problem can be solved by incorporating an insertion/extraction mechanism for them. Finally, we the finishing touches to our project (like integrating dashboard components) as necessary

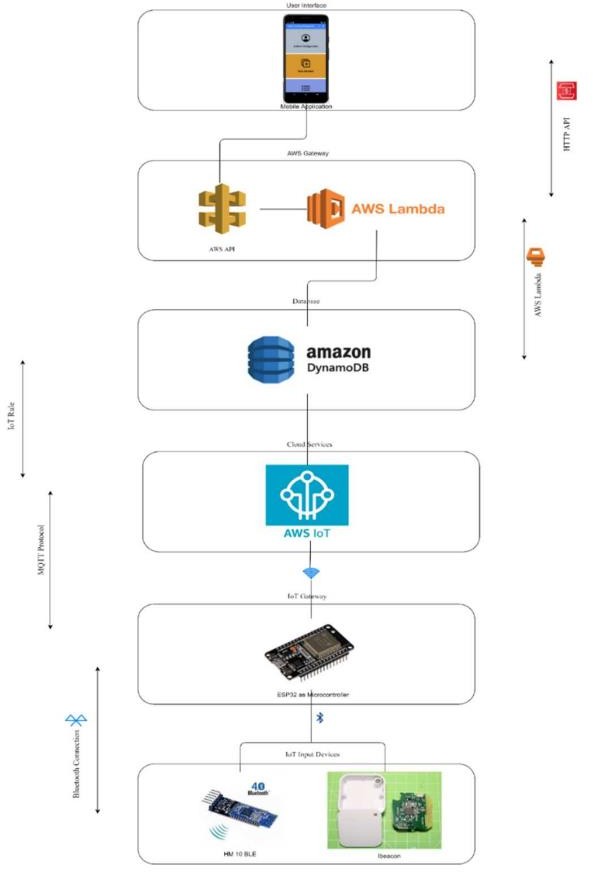


Fig 5.1 Integratinng the modules

*Dept. of ISE, BMSIT 2022-23* 26

**CHAPTER 6**

**CONCLUSION**

##### 6.1 CONCLUSION

In this project, we have proposed the implementation of iBeacon based smart shelves for retail stores based on IoT devices. It primarily addresses the issues related to retail stores. We critically evaluated the similar systems and justifies the use of IoT devices and other infrastructural components using a systematic approach, and finally proposed the architecture and recommendations for building the iBeacon based smart shelves for retail stores. Future developmentcan include automatic recognition of the products, indoor mapping navigation function by using ARtechnology and push notification to notify the users directly. Moreover, iBeacons can also be utilized for proximity marketing.

We built a simple, flexible and scalable IoT shelf system. Low cost electronic components were used. We designed and developed the needed software components to have a proof of the prototype concept of our system. We have tested the developed system, and test shows that the concept can beused as a solution to manage shelf inventory and stock replenishment by improving operational efficiencies. Using this system, we can also improve shopper experience, and assure on shelf availability of products. Our system does not use third party servers or components like well-known IoT technology components (MQTT clients and brokers). The sensor data acquisition circuit can be powered by accumulators for mobile shelves. MySQL database can be easily changed to a noSQL database for scalability to reach Big Data size. Just the data model must be changed in the web application. Also, additional custom statistics and data analysis can be developed if needed.

Image processing really seems to be a boon to CPG and Retail organizations. It is an almost zero investment clean technology solution that can be implemented almost immediately. Beyond CPG, many other implementation areas are also possible. For example, one of the garment manufacturers was interested if they can monitor the inventory levels of garments both folded on a shelf and also hung on hangers

**REFERENCES**

1. Peralta, E. Iris, and M. David Spooner. “History, origin and early cultivation of tomato (Solanaceae).” Genetic improvement of Solanaceous crops, vol. 2, pp. 1-27, 2006.
2. Moshou, Dimitrios, C. Bravo, Jonathan West, Stijn Wahlen, Alastair McCartney, and Herman Ramon. “Automatic detection of yellow rustin wheat using reflectance measurements and neural networks.” Computers and electronics in agriculture, vol. 44, no. 3, pp. 173-188, 2004.
3. Zhang, Yan-Cheng, Han-Ping Mao, Bo Hu, and Ming-Xi Li. “Features selection of cotton disease leaves image based on fuzzy feature selection techniques.” In: proceeding of the ICWAPR’07 International Conference
4. LeCun, Y., Bengio, Y., Hinton, G., 2015. Deep learning. Nature 521, 436–444.
5. Hinton, G., Deng, L., Yu, D., Dahl, G.E., Mohamed, A.R., Jaitly, N., Senior, A., et al., 2012.Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups. IEEE Signal Process Mag. 29 (6), 82–97.
6. Carranza-Rojas, J., Goeau, H., Bonnet, P., Mata-Montero, E., Joly, A., 2017. Going deeper into the automated identification of Herbarium specimens. BMC Evol. Biol.Yang, X., Guo, T., 2017. Machine learning in plant disease research. Europ. J. BioMed. Res. 6–9.
7. Lee, S.H., Chan, C.S., Wilkin, P., Remagnino, P. 2015. Deep-plant: Plant identification with convolutional neural networks. 2015 IEEE Intl Conf. on Image Processing, pp. 452–456.
8. S. Liang and W. Zhang, ‘‘Accurate image recognition of plant diseases based on multiple classifiers integration,’’ in Proc. Chin. Intell. Syst. Conf., vol. 594, 2020, pp. 103–113, doi: 10.1007/978-981-32-9698-5\_13.
9. S. M. Jaisakthi, P. Mirunalini, D. Thenmozhi, and Vatsala, ‘‘Grape leaf disease identification using machine learning techniques,’’ in Proc. Int. Conf. Comput. Intell. Data Sci. (ICCIDS), Feb. 2019, pp. 1–6, doi: 10.1109/ ICCIDS.2019.8862084.
10. K. Yang, ‘‘Development and validation of a deep learning algorithm for the recognition of plant disease,’’ in Proc. IEEE 21st Int. Conf. High Perform. Comput. Commun., IEEE 17th Int. Conf. Smart City, IEEE 5th Int. Conf. Data Sci. Syst. (HPCC/SmartCity/DSS), Aug. 2019, pp. 1951– 1957, doi: 10.1109/HPCC/ SmartCity/DSS.2019.00269.
11. A. Waheed, M. Goyal, D. Gupta, A. Khanna, A. E. Hassanien, and H. M. Pandey, ‘‘An optimized dense convolutional neural network model for disease recognition and classification in corn leaf,’’ Comput. Electron. Agricult., vol. 175, Aug. 2020, Art. no. 105456, doi: 10.1016/ j.compag.2020.105456.
12. Q. Tian, J. Li, and H. Liu, ‘‘A method for guaranteeing wireless communication based on a combination of deep and shallow learning,’’ IEEE Access, vol. 7, pp. 38688–38695, 2019, doi: 10.1109/ACCESS.2019.2905754.
13. B. M. Aslahi-Shahri, R. Rahmani, M. Chizari, A. Maralani, M. Eslami, M. J. Golkar, and A. Ebrahimi, ‘‘A hybrid method consisting of GA and SVM for intrusion detection system,’’ Neural Comput. Appl., vol. 27, no. 6, pp. 1669–1676, Aug. 2016.
14. A.-C. Enache and V. Sgarciu, ‘‘Anomaly intrusions detection based on support vector machines with an improved bat algorithm,’’ in Proc. 20th Int. Conf. Control Syst. Comput. Sci., May 2015, pp. 317–321.
15. A.-C. Enache and V. V. Patriciu, ‘‘Intrusions detection based on support vector machine optimized with swarm intelligence,’’ in Proc. IEEE 9th IEEE Int. Symp. Appl. Comput. Intell. Informat. (SACI), May 2014, pp. 153–158.
16. N. R. Bhimte and V. R. Thool, ‘‘Diseases detection of cotton leaf spot using image processing and SVM classifier,’’ in Proc. 2nd Int. Conf. Intell. Comput. Control Syst. (ICICCS), Jun. 2018, pp. 340–344, doi: 10.1109/ ICCONS.2018.8662906.
17. P. B. Padol and A. A. Yadav, ‘‘SVM classifier based grape leaf disease detection,’’ in Proc. Conf. Adv. Signal Process. (CASP), Jun. 2016, pp. 175–179, doi: 10.1109/CASP.2016.7746160.
18. A. K. Jain and R. C. Dubes, Algorithms for Clustering Data. Englewood Cliffs, NJ, USA: Prentice-Hall, 1988.
19. MacQueen,‘‘Some methods for classification and analysis of multivariate observations,’’ in Proc. 5th Berkeley Symp. Math. Statist. Probab., vol. 1, 1967, pp. 281–297.
20. M. Alhawarat and M. Hegazi, ‘‘Revisiting K-means and topic modeling, a comparison study to cluster arabic documents,’’ IEEE Access, vol. 6, pp. 42740–42749, 2018.