PROJECT REPORT
ON
COAL FIRE ESTIMATION
(MINE FIRE)

D E V E L O P E D A T

UNDER THE GUIDENCE OF
Shri Ajay Khalkho
(Principal Scientist,Mine Fire Division,CIMFR,Dhanbad)

DEVELOPED BY

Deepanshu Jayswal
Roll No: 1728263

B a c h e l o r o f T e c h n o l o g y
(Session: 2017-2021)

Preface

Indian coal mines have a historical record of extensive fire activity for past 140 years (Raniganj Coal field, 1865). The coal mine disasters in India revealed that fire and explosion contributed to about 41% of total fatalities. The incidence of fire in Indian coal mines IS mainly due to spontaneous combustion of coal It presents a serious safety hazard to the mine personal Sometimes, even a small incidence of fire can lake a heavy toll in terms of loss of production, machines and fire fighting equipments, upon requiring a sealing of a large section of mine or entire mine. This also contributes to global problems such is loss of coal reserve and environmental pollution which will adversely affect economics of mining directly or indirectly.

The software **Coal Fire Estimation** (Mine Fire) is based on this underground mine fire, This software is being developed with the intention to provide a detection of this underground mine fire and the developer **Mr.Deepanshu Jayswal** has performed a satisfactory job for the last two months of this project.

Remarks________________

(Ajay Khakho)
Principal Scientist
Mine Fire Division
CIMFR, Dhanbad

CSIR - CENTRAL INSTITUTE OF MINING AND FUEL RESEARCH

Barwa Road, Dhanbad - 826015

Certificate

This is to certify that the Project entitled **"Coal Fire Estimation"** submitted by **Mr Deepanshu Jayswal, Roll No- 1728263,** Department of Computer Science, KIIT University, Bhubaneswar in fulfilment of a summer internship at CSIR - CIMFR is an authentic work carried out by him under my supervision and guidance.

To the best of my knowledge, the matter embodied in the project has not been submitted to any other University/Institute for the award of any degree.

Date: 4 *th* July 2019

AKhalkho
04.07.2019

(Ajay Khalkho)
Principal Scientist
Mine Fire Division
CIMFR, Dhanbad

Acknowledgement

First of all I would like to give special thanks to **Director of CIMFR, Dr. P. K. Singh**, who has provided me the opportunity to do my minor project work in CIMFR, Central Institute of Mine and Fuel Research, Dhanbad

I express my deep senses of gratitude to **Shri Ajay Khalkho (Principal Scientist, CIMFR, Dhanbad)** under whose guidance the present project work has been carried out successfully am highly indebted to him for his valuable and sincere guidance and constant encouragement during the development of this project work. I again thank him for providing his precious time to me in the completion of my project. I shall remain ever grateful to him, who unselfishly inculcated in me the spirit of hard work and help me in each and every aspect of this project. He had always provided me a new tricks and techniques to me for solving my project's problems.

I am also thankful to Mr. D. Kubhakar, Scientist-in Charge, HRD and all the staffs of HRD division for providing all the necessary help and facilities for carrying out the work.

Further, I am thankful to Prof. Samaresh Mishra, Dean of KIIT University, Bhubaneswar, who provided me with this opportunity and sent me CIMFR,Dhanbad to do my 3rd Semester Minor Project Work.

Deepanshu Jayswal

Declaration

I **Deepanshu Jayswal** here by declare that the project submitted to the university has been entirely programmed by me to fulfill the requirement of **B-Tech 4th Semester's Minor Project** of **Roll Number 1728263** of **KIIT University, Bhubaneswar.**

I Declare that this project has been completed within the given time slots and facilities provided by **CIMFR, Dhanbad** where my project training was conducted.

I also declare that the project work sent to the university has not been produced or presented before any other university for any kind of degree and it is genuine.

Deepanshu Jayswal
Roll No: 1728263

INDEX

ABOUT CIMFR

CSIR-CENTRAL INSTITUTE OF MINING AND FUEL RESEARCH (CIMFR)DHANBAD, a constituent laboratory under the aegis of Council of Scientific and Industrial Research (CSIR), New Delhi aims to provide R&D inputs for the entire coal-energy chain encompassing exploration, mining and utilization. The laboratory also strives to develop mineral based industries to reach the targeted production for a country's energy security and growth with high standards of safety, economy and a cleaner environment. In view of the National Missions recently declared by the Government of India, CIMFR has realigned its vision, missions and policies and also redefined targets for short and long term. This would promote rapid sustainable national techno-economic growth with equal emphasis on self-sustenance. CSIR-CIMFR is located in the town of Dhanbad, known as coal capital of India of Jharkhand state of India. It is strategically situated in the Damodar basin of Eastern part of the country which is endowed with rich coal deposits and hosts several large mineral based industries.

Vision

To Be An Internationally Acclaimed Mining And Fuel Research Organization.

Mission

To Develop And Deliver Sustainable Cutting Edge Technologies For Social Upliftment And Industrial Advancement.

Mandate

To innovate safe and economically viable technologies, to make the best use of the available knowledge base and to transfer know-how in line with National Missions and the Dehradun Declaration with focus on -

- Maximum recovery of fossil fuels, ores and minerals;
- Clean coal initiatives with optimum and effective utilization of low-grade coal along with waste management;
- know-how and R&D service support to the mining industry in general and coal industry in particular from “Mine to Market”;
- scientific support to strategic and other important sectors like atomic energy, defence, railways, hydro-electric projects, archaeology, city planning, agriculture etc.;
- Mass mining of deep-seated coal and mineral deposits;
- The exploitation of difficult coal seams;
- Environment protection including monitoring and mitigation of greenhouse gas emissions in mining and coal-based industries;
- Serve Societal Needs;
- Facilitation of mutually beneficial interaction between industry and society for perpetual techno-economic and societal growth;
- Strengthening linkages with academic and research centres of excellence, industry and other relevant institutes for exchange of knowledge and transfer of technology; and Generation and dissemination of database and know-how as and when required.

Milestones:

Central Fuel Research Institute (CFRI) -1946,

Central Mining Research Institute -1956,

CSIR-Central Institute of Mining and Fuel Research-2007

Core R&D Strengths:-

- Coal and Non-coal Mines, and Civil Sector
- Fuel Sector
- Environmental Management
- Testing, Analysis and Calibration Services

Unique Facilities:-

- National facilities for FLP, Intrinsic Safety, Explosive & Mining safety equipment testing.
- Testing facilities for coal & rock properties, environmental parameters, coal processing, monitoring of U/G mine's geo-mining conditions.
- High temperature & pressure single tube fixed bed reactor, & pilot plants for coal processing.
- High-performance computing for large scale geo-mining modelling and simulation
- Scanning Electron Microscope
- Coal-to-liquid plant

National Development:

- Eco-friendly mine planning and design, and clean coal technologies.
- Cost-effective mining technologies for enhancing productivity and safety.
- Design of safe excavation method for tunnels, railways and hydroelectric projects.
- Quality evaluation of coal for effective utilization.
- Utilization of fly ash for U/G mine filling.
- Coal dust collecting and briquetting system.
- Sensor-based landslide monitoring and detection system.
- Pollution monitoring and control, waste management in mining areas, mine water reclamation, & rural watershed management.
- Energy audit for optimum utilization of power and innovation of different forms of fossil fuel.

Mining & Fuel Industries:

- Biodegradable and eco-friendly dust suppressant chemicals for mining areas.
- Fly ash utilization for stowing in underground mines, reclamation dumps and bricks.
- Coal carbonization technology.
- Technology for soft coke preparation.
- Coal combustion technology.
- Coal liquefaction technology.
- Coal gasification technology.
- Development of chemicals from coal.
- Coal preparation for the power sector and coke ovens.
- Wireless information and safety system for mines.
- Environmental monitoring system for coal mines.
- Design of highwall mining.
- Strata movement warning system of U/G mines.
- Trapped miner locator.
- National facilities for testing of FLP, intrinsic safety, explosive, mining safety equipment; coal & rock properties, environmental parameters, coal processing, mineral processing etc.

Strategic Area:

- Tunnelling and space technology.
- Mine planning and EIA for construction of Karwar sea water barricade of Indian Navy.
- Method of working and EIA/EMP for uranium mines of UCIL.
- Worked with DRDO on Multi-Feed Gasification of High Ash Coal, Biomass and MSW Blends.

Societal Mission:

- Mine water reclamation technology for safe drinking water.
- Training to the villagers for socio-economic benefits, viz. cultivation of medicinal plants, revegetation, carbon sequestration, value-added products etc.

Mining Technology

Coal and minerals involves the design of openings of different natures and, from the stability point of view, these openings may be divided into, mainly, three categories: short, medium and long term stable. These openings are made under widely varying geo-mining conditions of the country ranging from strong and elastic rock mass of igneous origin to brittle and fragile strata of sedimentary origin. Further, it is not straight-forward to apply a foreign technology for extraction of our minerals and coal due to uniqueness of Indian rock mass. Since last sixty years, CIMFR is studying engineering rock mechanics and ground control issues for optimal design of underground mining openings under Indian geo-mining conditions. Considerably large number of investigations has successfully been completed by the institute resulting development of different valuable indigenous norms, formulations and technologies. Extensive field investigations/testing and simulation on numerical and physical models are the basic tools for these developments. Today, CSIR-CIMFR is comfortably seating on the precious knowledge base and technical treasure to take up the arising future problems mining under difficult geo-mining conditions.

CSIR-CIMFR has developed considerable expertise and knowledgebase of Indian geo-mining conditions and developed a number of widely practiced valuable formulations. Ten important formulations of CSIR-CIMFR are:

- Rock Mass Rating (RMR) for roof strata characterization and support design.
- Rock-mass failure criterion for Indian coal measure formations.
- Safe limit of vibration for the security of surface structures due to opencast blasting
- Empirical formulation for pillar strength estimation.
- Empirical models to estimate range and amount of mining induced stress.
- Analytical formulation for *in situ* stress estimation.
- Subsidence prediction models for single and multi-seam mining conditions.
- Formulation for the optimal size of rib/snook in mechanised depillaring.

- Formulation for Cavability Index to estimate overlying strata behaviour. Roof Bolt Based Breaker Line Support (RBBLS) design norm for CM based mechanised depillaring.

Applications of basic rock mechanics principles along with important inputs of rock mass behaviour and understanding of geo-mining conditions have resulted following important developments, which are being widely practiced by the mining industry:

1. Cable-based depillaring of total thickness of a thick coal seam in single lift.

2. Short-wall mining technology for extraction of locked-up coal in pillars.

3. Resource evaluation for coal bed methane and coal mine methane.

4. Technology for extraction design of locked-up coal by Highwall Mining.

5. Staggered development of a thick coal seam (already developed along roof horizon) for full height working in single lift by blasting gallery method.

6. Underpinning based simultaneous extraction of contiguous sections of a thick coal seam consists of the weak and laminated parting.

7. Efficient blasting and fragmentations techniques.

8. Neutralizing the excess pressure differential across the sealed-off for improvement of climatic conditions of underground workings.

9. Hard roof management with induced caving of hard roof.

10. Fly-ash backfilling technology for u/g void stabilization.

INTRODUCTION

Mine Fire is a serious problem for the coal mining industry and will continue to mine safety endangering human lives, losing valuable natural resources, making mining difficult or impossible and polluting environment from the evolution of toxic gasses and fumes.

The development of virtual mine fire detection, prevention, control and combining technology will give a quantum jump for the Indian mining industry to survive in this liberalized and competitive global economy. In the last few years, the application of information technology has achieved considerable success in many parts of the mining industry. It also is used in the various aspects of the mine fire. Some research workers have already applied information technology for mine fire detection, prevention and control.

This paper reviews the status of application of information technology at present and future needs of information technology in mine fire detection, prevention and control. Infrastructure needs and procedure to develop an information management system are described.

Coal mining in India started dates back to 1774, when Sumer & Heatly first discovered coal deposit Raniganj coalfield. Fire in coal mines has a historical record of extensive fire activity for over 140 years back (Raniganj coal field, 1865). The mine disasters in India revealed that fire and explosion contributed to about 41% of total fatalities. It presents a serious safety hazard to mine personnel and economic point of view even a small incident of fire can take a heavy toll in terms of loss of production, machinery and fire fighting material, often requiring sealing a large section of mine or the entire mine. Besides these mine fire also essentially contributes to global problems such as loss of important coal reserve and environmental pollution etc. A tragic fire incident at New Kenda colliery of ECL (1994), resulted in a loss of 55 lives and closure of the mine demanded more research on assessment and control of mine fire.

The high risk associated with mine fire has brought about a number of innovations in the field of detection, prevention, control and combating the mine fires. The various laboratory methods and techniques dealing with mine fire are now well developed in monitoring, sealing, cavity filling, inertisation, ventilation and extraction. In the last few years, the application of information technology has been developing rapidly and finding more successful applications in many parts of mining. It is largely been confined to those operations, which tend to build applications, which can aid decision-making, requiring empirical knowledge, subjective judgement, experience as well as certain information. The main problem, presently faced by mine management, is to acquire the in situ real-time data, filter the relevant information from these data, and, implement an appropriate decision within the desired period to optimise the mine productivity and safety. In the era of information technology, it is to consider transforming this knowledge into computer programmes, capable of analysing the mine fire problem and recommending control strategies to allow the safe extraction of coal.

Application of Information Technology for Mine Fire

From early 1970, many scholars have been working on the areas using a digital computer to simulate underground fire.

Nordan et. al., (1979) developed a one dimensional model including the complete differential for conservation of oxygen water, energy and for the reaction rate of oxygen with coal for mass transport both convection and diffusion were included for heat transport both convection and conduction. Although this model was complete, the influence of moisture was not incorporated.

Stefanove et al., (1984) formulated a computer programme (VENT-4) to solve the complex simulation of a mine ventilation network during an open fire. They have been used unsteady state conditions in the system to give the result from the following factors ie; temperature changes in the fire zone, gradual filling of ventilation network with combustion gases and it locate the heat transfer between these gases & the rock

surfaces.

Deliac et al., (1985) gave a module PC fire, which works as an additional module of PC Vent. It is based on the equations derived by Simode and Vielledent to calculate the effects of mine fire in a tunnel on the temperature, the mass flow and the pressure variation. In situations whereby evaporation can be neglected and the fans are not stopped, it has been found to converge in most cases the first results seems very permissive. The PC fire should be considered as a tool for training for prevention purposes rather than making decision when fighting against a sire.

Schmal, (1987) developed TNO model, which was one dimensional convective and diffusive transport model including chemisorptions of oxygen to the coal, oxygen depletion in the pile, heat generation as a result of the chemisorptions, cooling and heating due to evaporation and condensation of water.

Husheng, (1988) developed a computer programme for calculation of thermal pressure and solution of the ventilation network. He used the temperature distribution of smoke flow down casting the fire seat under fire circumstances.

Yihui et al., (1988) formulated a mathematical model (UNIVAC-1100/10) on the characteristics difference method is used directly to solve fluid dynamic equations in the ventilation network during underground fire. The model works in the principles of diffusion dynamics, the distribution states of noxious gases in the mine have been analysed by the method of characteristics.

Dziurzynski et al., (1988) derived a mathematical model to show the behaviour of the ventilation system caused by a fire make provisions for emergency plan. This model has been based upon the combustion of fuel in the airways, unsteady exchange of heat between the fire, airflow and surrounding rocks i.e., the air flow in mine ventilation network and theory of heat sow in the rocks. The model facilitates a graphical presentation of the spread of the combustion products throughout network.

USBM developed MFIRE to calculate the flow rates for the airway with the fire source are used or input into the gravity current model (Greuer, 1977, 1987; Edward, 1981 and Chang et. al., 1990). Fire is modelled here as constant sources of heat with smoke and fumes being passive scalar contaminants. The model results for up-streams & down-streams heat flows and fume concentration gravity current breakdown are feed to MFIRE.

Chang et al., (1990) simulated a computer simulation programme namely, MFIRE, which performs normal ventilation network planning calculation and dynamic transient state simulation of ventilation networks under a variety of conditions including fires.

John et. al., (1985) and Jones, (1991) developed a 3-D finite volume code namely, FLOW-3D for simulating turbulent flow and heat transfer by the UK atomic energy authority. The model is based on non orthogonal body fitted grid which is arranged in a structured blocks allowing complex geometrical features to modelled and solves the governing equations for mass, momentum and energy conservation along with modelled turbulence equation.

Zhu et. al., (1991) formulated a mathematical model for air flows and self-heating to goaf is in its early stages.

Denby et. al., (1992) developed a knowledge based expert system namely EESH for the risk control of spontaneous combustion in underground coal mining. The system is capable of combining the certainty factors to provide a quantitative assessment of self-heating risk of coal in a specific mining situation. It is also able to identify the measure factors contributing the heating potential strategies against the possible risk are recommended using specialist knowledge based on the information integrated from the user and a relational database.

Lea, (1991) developed a computational Fluid Dynamics (CFD) modelling to demonstrate potential for simulating the near fire low CFD techniques by the large number of grid cells necessary to describe the roadway geometry to simulating the phenomena close to a fire.

Smith et. al., (1995) developed SPONCOM to aid in the assessment of the spontaneous combustion risk of an underground mining operation. A prior knowledge of the spontaneous combustion risk of coal and factors that increase risk can be useful in the planning and development of proactive monitoring ventilation and prevention plans for the mining operation. It determines the coal relative spontaneous combustion potential based on the coals proximate, ultimate analysis and heating value. Then it evaluates the impact and coal properties, geological & mining condition and mining practices on the spontaneous combustion risk of mining operation.

Saghafi et. al., (1995) derived a mathematical model of air, water and heat flow in a porous medium was developed and applied to an underground longwall goal for predicting self heating behaviour of underground mines.

Panigrahi et al., (1996) developed the software VENTSYS to combat heat and humidity problems in India. This software is a very useful tool for ventilation network analysis and planning of underground environment.

Prakash et al., (1996) has been used remote sensing GIS techniques for estimating temperatures of ground surface above the subsurface coal fire.

Tripathi et al., (1996) formulated an algorithm based on some recent fire indices, which provides some significant information relating to fire hazards within a coal mine including extent of a fire burning within a scaled environment.

Saghafi, (1997) simulated a new mathematical model of spontaneous combustion applied to longwall mining. It simulates the process of heat generation and heat transport within a goaf. The heat is generated by low temperature oxidation of carbonaceous material as well as adsorption of water vapour solid material in the goaf. The model was applied to predict the temperature rise in longwall goaf where a particular ventilation layout was adopted to control both gaseous emission and spontaneous combustion hazards.

Roy et. al., (1997) formulated computer program CLIMA-S for prediction of transient climatic conditions in the underground airway is developed using well-established psychometric and thermodynamic relations.

Zoltane et. al., (1999) developed a PC based information system namely, COALMAN in a framework of Dutch-Chinese

project for one of the most important coal mining areas in Hu Ningxia Hui Autonomous Region to integrate data and models in to management tool. It will be better tools for monitoring the fires, managing the fire lighting and to work out proper fire fighting and prevention plan.

Fierro et. al., (2001) Modified the TNO one-dimensional model to predict the spontaneous heating of stock piled coals. The model is based on conservation of mass oxygen, heat flow and gives as a result time and site profiles oxygen concentration in air, adsorbed oxygen concentration and calorific loss. It includes the influences of parameters such as free convection due to the temperature rise of the pile and additional forced convection (wind speed direction). The model generally predicts very well, the order of magnitude of time and the site of spontaneous combustion. The model supports the calculation of calorific losses when the hot spots are developed. The coefficient of heat losses also calculated.

Akgun et. al., (2001) developed a theoretical two-dimensional unsteady model to predict the existence of hot spots of a stockpiles. It is capable of predicting the realistic ignition times of the order of a month, at realistic pile heights of 2 or 3 m.

Shay et. al., (2002) developed a computer program EXPLO based on Ellicott's extensions of Coward's diagram for quick and accurate assessment of explosibility of air mixture build up in case of control of active fire. The advantage of this software is the first and reliable analysis of data and reduction in risk of misinterpretation of graphical information.

Need of Computerisation of Minefire

In the era of information technology, the world is experiencing major technological revolution, centered on information and its communication. Today, business excellence, economy and safety are governed by accurate and efficient decision making. Under the present knowledge and technology explosion scenario, information technology is the strong arms to such an environment. It works on the principle i.e. a set of tools that can help to provide the right people with the right information at the right time. It enables expeditious and efficient data capturing, assimilating the same data, analyzing and decision making in a fraction of seconds by networking keeping away time and distance.

Hierarchy of information, from data to actions.

More direct objectives are to maximise productivity, minimise unit cost of production (including maintenance), maintain safe and reliable operation, comply with safety and environmental standards, and ensure a long-term future by strategic planning. Timely and accurate information is critical to support planning and control functions.

Sometimes innovative technology to improve efficiencies and reduce costs is available to the industry. However, the infrastructure needed for integration of these technologies and handling the data have lagged the actual development of the technology. In the present era of computerisation, mining methodologies and processes are being modified through the introduction and use of new technologies. So the solution in the age of application of information technologies is to consider transferring this knowledge into computer programmes capable of analysing a mines spontaneous heating problem and recommending control strategies to allow the safe extraction of coal. A knowledge-based expert system is an important practical aspect of artificial intelligence (AI) can meet this demand.

The Indian mining industry will survive by innovating and optimising the operations with the help of the rapidly emerging technological opportunities and realise the economic benefit of improved efficiencies throughout the organisation. The development of virtual mine fire detection, prevention, control and combating technology will give a quantum jump for Indian mining industry to survive in this liberalised and competitive global economy. Considering the coal will remain the prime source of energy for future generations, the concept of this technology and suitable R&D investment in this area may give more economical and safe extraction of coal, loss of natural resources and cut off global environmental pollution,

The most of Indian coal mines are very old, which worked earlier unscientifically and are without record. In the national level no researcher has tried to develop an expert system according to Indian scenario. The expert systems, which are developed earlier by different countries, are not suitable to our coal mines and soul remains at an elementary level of design and further modification & refinement are required. In this system the status of fire in any working or abandoned coal mining area is not involved. This may be useful for the mine during planning and design stages. These models are not considering the mines, which are now in productive phase or abandoned stage. With the above discussion the development of a new expert system in Indian scenario will provide an effective approach to the minimisation of spontaneous combustion incidents. This expert system will uses available information from the mine officials to make a decision based on a series of rules provided by a programme, which will be interactive, user friendly and inexpensive method of conveying expert advice.

Need for Infrastructure

IT investments by mining companies have made often failed to provide the promised benefits due to the haphazard way of investment. So careful planning, more methodical information of risks, quantification of risk and mitigation of those risks along the same lines that are applied to other investments in the mining industry would pay dividends for mining companies.

The basic use of an efficient IT system is to enable real-time access to data from all mining and its coordinating & supporting processes. The data are integrated into a central data repository, to be assessed from anywhere in the organisation. Accuracy and timeliness of data are improved and full integration of mine management information system becomes possible.

The infrastructure needs of an efficient IT system of a mining organisation include:

- On-board machine monitoring, control and positioning systems.
- An open architecture object-oriented mine modelling and GIS (Geographical Information System).
- An integrated database system that permits ready access by operations, maintenance or management groups using flexible, application-specific software.
- A comprehensive mine planning system utilising the information and control capabilities of the mining system.
- Bi-directional mobile communications system with adequate bandwidth (Peck et. al. & Zoltan, 2001).

Scope for Application of Information Technology on Mine Fire

The first step in implementing the next generation of mine planning and operational control systems is to specify overall information system architecture, one that is open, flexible, and scalable. The key should be to develop such a system, which allows interactions between applications to foster integration. System design should be both top-down and bottom-up.

Nowadays few software is available in the market for different aspects of the mine fire. Among which SPONCOM, COALMAN and TNO modified model is better as compared to other. SPONCOM software is very much useful for mine planning and design according to spontaneous combustion risk but for Indian context it needs some change in its structure, so that it will help for our mining industry. COALMAN will be better tools for monitoring the fires, managing the fire fighting and to work out proper fire fighting and prevention plan. TNO modified model is very suitable for determination of spontaneous combustion risk in coal stockpile. This model will not efficiently work in different climatic conditions, so further modification is required in this field. In the present world there is no single software, which will be useful for different aspects of mine fire. So different modules of software are required for detection, prevention, control & combating aspects and integration of these modules of mine fire to make comprehensive package.

The mining industry requires skilled IT manpower to upgrade knowledge of correspondence courses, attending seminar & conference and discussing different matters through email, chat rooms or web conferencing. Therefore an environment is required for mining industry people to use this technology in future. Intranets, with restricted access, reside behind company firewalls, supporting and enabling intra-company processes. Intranet can also displace the traditional practice of paper based information distribution, increasing the efficient use of emergency plan in case of mine fire. Extranets, secure hybrids of internet and intranet technologies with restricted access to the community membership, support and enable inter-company processes and extend the enterprises beyond its traditional boundaries.

Need of Coal Fire Estimation Software

In the era of information technology, the world is experiencing a major technological revolution, centered on information and communication techniques. Under the present knowledge and technology explosion scenario, information technology is the strong arms to such an environment. It works on the principle i.e. a set of tools that can help providing the right people with the right information at the right time. In the present era of computerisation, mining methodologies and processes are being modified through the introduction and use of new technologies. So, the solution in the age of application of information technologies is to consider transferring this knowledge into computer programmes capable of analysing a mines spontaneous heating problem and recommending control strategies to allow the safe extraction of coal. The development of virtual mine fire detection, prevention, control and combating technology will give a quantum jump for Indian mining industry to survive in this liberalised and competitive global economy.

Now a days few software available in the market for different aspects of mine fire. Among which SPONCOM, COALMAN and TNO modified model is better as compared to other.

SPONCOM software is very much useful for mine planning and design according to spontaneous combustion risk but for Indian context it needs some change in its structure, so that it will help for our mining industry. COALMAN will be better tools for monitoring the fires, managing the fire fighting and to work out proper fire fighting and prevention plan. TNO modified model is very suitable for determination of spontaneous combustion risk in coal stockpile. This model will not efficiently work in different climatic conditions and hence further modification is required in this field. In the present world there is no single software, which will be useful for different aspects of mine fire. So different modules of software are required for detection, prevention, control & combating aspects and integration of these modules of mine fire to make comprehensive package. This software will help for the detection of mine fire from gas analysis data. Earlier researchers had not developed this type of software for assessing the early detection of heating in working or sealed off panels of a mine.

The basic use of an efficient IT system is to enable real time data access from all mining areas, coordinate and support the processes. The data are integrated into a central data repository, to be assessed from anywhere in the organization. Accuracy and real time data are improved and full integration of mine management information system becomes possible.

DIFFERENT INDEX RATIOS

GRAHAM'S RATIO (1921): $100CO/(0.265N_2-0_2)$
where N2= $(100-CH-CO-C0_2-0_2-H_2-C_2H_2)$

YOUNG'S RATIO: $100(CO_2-0,03)/(0.265N_2-0_2)$

MITCHEL RATIO: CO/CO_2

BANERJEE RATIO: 6(CO2+CO+C2H2+2C2H4)/ 2(0.265N2-02-CO, CIL+C2112+112-CO) where N2=(100-CH4-CO-CO2-02-H2-C2H2-C2H4)

DESORBED HYDROGEN INDEX: 1.01*(CH4+C2H2+C2H4)-CH4 * 1000

(CH+C2H2+C2H4 +0.01 WR (H.L.WILLET 1951)= CO EXCESS N2+CO2+COMBUSTIBLE

LITTON'S RATIO:

1/3[(CO)s R3 -3/20 -1/27 where Rs=(100-4.77402-CH4-C2H2 and (CO) s=CO in PPM

JTR: (CO2+0.75CO-0.25H2)/(0.265N2-02) Where N2=(100-CHA-CO-CO2-02-H2-C2H2-C2H4)

MORRIS RATIO(MR): N2/(CO+CO2)

RI/RE: (1-3.8302)/N2]/CO DEF 0.265(N2-3.8302)/(CO+CO2)

ACETYL/ETHYLENE: C2H2/C2H4 HYDROGEN/ METHANE= H2/CH4

Tools & Platforms Used

Software Component

OPERATING SYSTEM : Microsoft Windows 10

LANGUAGE USED: Java

DATABASE USED: Text

OTHER PACKAGES: JFreeChart
Ms Excel
Adobe Photoshop 6.0

Hardware Component

PROCESOR: Intel Core i5

PRIMARY MEMORY(Ram): 8Gb

SECONDARY MEMORY(Hdd): 1Tb

KEYBOARD: Full Size Multimedia Keyboard

MONITOR: Full HD (14 Inch)

PRINTER: HP Laser Ink jet printer

BACKUP MEDIA: Hard disk Compact disk

Analysis

PROCESS MODEL USED

Though Classic Life Cycle model is the most popular and widely used model but the need of "Coal Fire Estimation" software is to provide some working version of the software with its development. So, as per the requirement I found "Incremental" paradigm is the best approach. Here is a brief description of the aforesaid paradigm :

The incremental model combines elements of the linear sequential model (applied repetitively) with the iterative philosophy of prototyping. Referring to figure the incremental model applies linear sequences in a staggered fashion as calendar time progresses. Each linear sequence produces a deliverable "increment" of the software. For example, word-processing software developed using the incremental paradigm might deliver basic file management, editing, and document production functions in the first increment; more sophisticated editing and document production capabilities in the second increment; spelling and grammar checking in the third increment; and advanced page layout capability in the fourth increment. It should be noted that the process flow for any increment can incorporate the prototyping paradigm.

When an incremental model is used, the first increment is often a core product. That is, basic requirements are addressed, but many supplementary features (some known, others unknown) remain undelivered. The core product is used by the customer (or undergoes detailed review). As a result of use and/or evaluation, a plan is developed for the next increment. The plan addresses the modification of the core product to better meet the needs of the customer and the delivery of additional features and functionality. This process is repeated following the delivery of each increment, until the complete product is produced. The incremental process model, like prototyping (Section 2.5) and other evolutionary approaches, is iterative in nature. But unlike prototyping, the incremental model focuses on the delivery of an operational product with each increment. Early increments are stripped down versions of the final product, but they do provide capability that serves the user and also provide a platform for evaluation by the user.

Incremental development is particularly useful when staffing is unavailable for a complete implementation by the business deadline that has been established for the project. Early increments can be implemented with few people. If the core product is well received, then additional staff (if required) can be added to implement the next increment. In addition, increments can be planned to manage technical risks. For example, a major system might require the availability of new hardware that is under development and whose delivery date is uncertain. It might be possible to pian early increments in a way that avoids the use of this hardware, thereby enabling partial functionality to be delivered to end user without inordinate delay.

FEASIBILITY STUDY

Previously, for the purpose of detection of “Mine Fire” manual system was used. Computerization of the above systein make it economical, technical and efficient. The above feasibility has to be decided with due care. The reason thereof is the inherent factor of project risk.

As per the requirement a GUI system has to be run. The organization mainly uses Windows 10 as individual as well as in the group with server. Further, the organization has systems that incorporate 20GB hard disk that is adequate to hold the data required using the new system. The systems in the organization also include Pentium IV or above processor that will provide fast processing of data and responses to the user.

JAVA Environment

Java is a general-purpose computer programming language that is concurrent, class-based, object-oriented etc.

Java applications are typically compiled to **bytecode** that can run on any Java virtual machine (JVM) regardless of computer architecture.The latest version is **Java 11**.

Below are the environment settings for both Linux and Windows. JVM, JRE and JDK all three are platform dependent because configuration of each Operating System is different. But, Java is platform independent.

There are few things which must be clear before setting up the environment

1. **JDK**(Java Development Kit) : JDK is intended for software developers and includes development tools such as the Java compiler, Javadoc, Jar, and a debugger.
2. **JRE**(Java Runtime Environment) : JRE contains the parts of the Java

libraries required to run Java programs and is intended for end users. JRE can be viewed as a subset of JDK.

3. **JVM:** JVM (Java Virtual Machine) is an abstract machine. It is a specification that provides runtime environment in which java bytecode can be executed. JVMs are available for many hardware and software platforms.

Data Flow Diagram

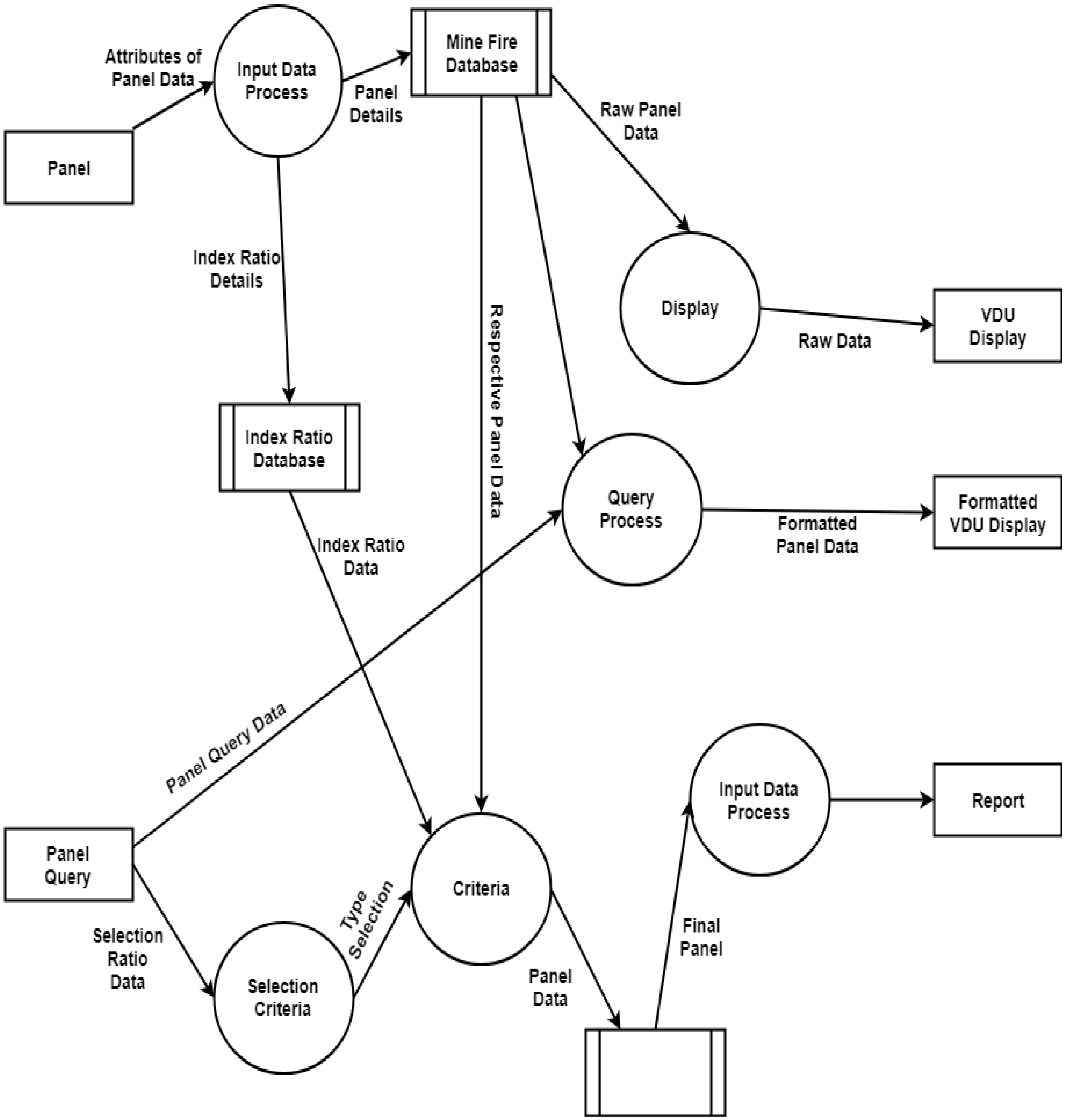
Information is transformed as it flows through a computer based system. The system accepts input in a variety of forms; applies hardware, software, and human elements to transform it; and produces output in an easy way.

As information moves through software, it is modified by a series of transformations. A data flow diagram (DFD) is a graphical representation that depicts information flow and that transforms that are applied as data move from input to output. The data flow diagram is also known as a data flow graph or a bubble chart.

The data flow diagram may be used to represent a system or software at any level of abstraction. Infact DFDs may be partitioned into levels that represents increasing information flow and functional detail. Therefore, the DFD provides a mechanism for functional modeling as well as information flow modeling.

A level 0 DFD, also called a fundamental system model or a context model, represent the entire software element as a single bubble with input and output data indicated by incoming and outgoing arrows respectively. Additional processes (bubbles) and information flow paths are represented as the level 0 DFD is partitioned to reveal more detail.

A level-1 DFD for the Coal Fire Estimation (Mine Fire) is given below:


Attributes of Panel Data
Input Data Process
Mine Fire Database
Panel Details
Panel
Raw Panel Data
Index Ratio Details
Respective Panel Data
Display
Raw Data
VDU Display
Index Ratio Database
Query Process
Formatted Panel Data
Formatted VDU Display
Index Ratio Data
Panel Query Data
Input Data Process
Report
Panel Query
Criteria
Type Selection
Selection Ratio Data
Selection Criteria
Final Panel
Panel Data

(Level-1 DFD)

Flow Charts

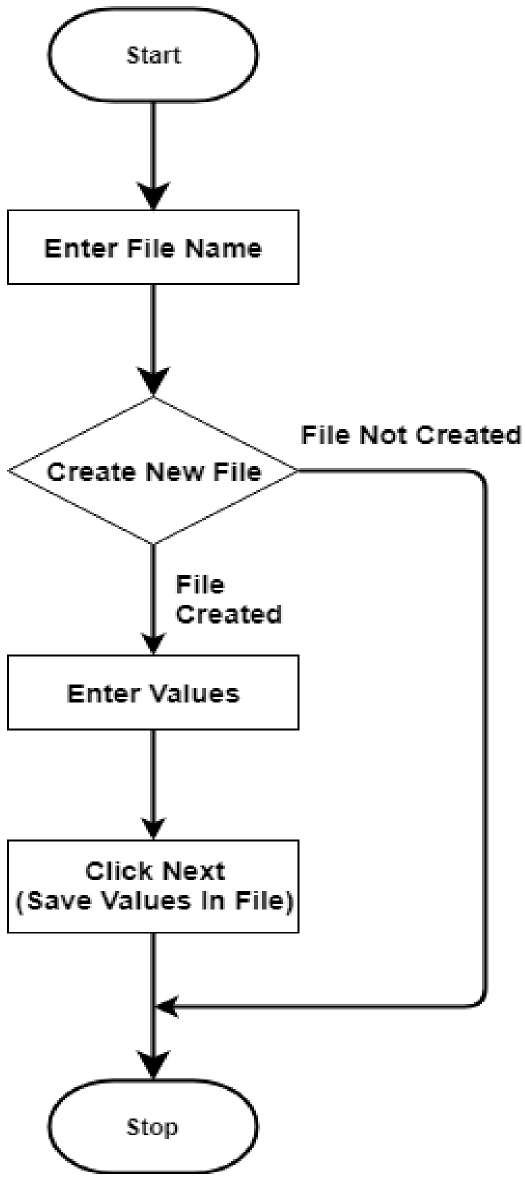


Fig. 1

Coal Fire Estimation

Programmed by: Deepanshu Jayswal

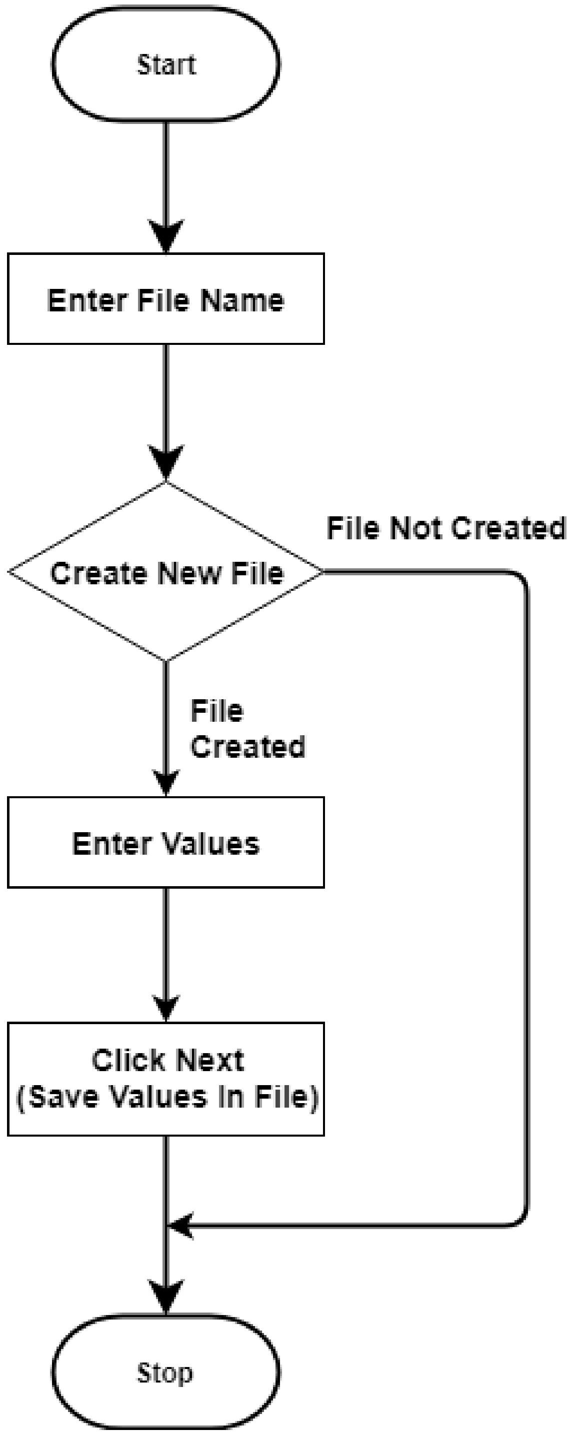
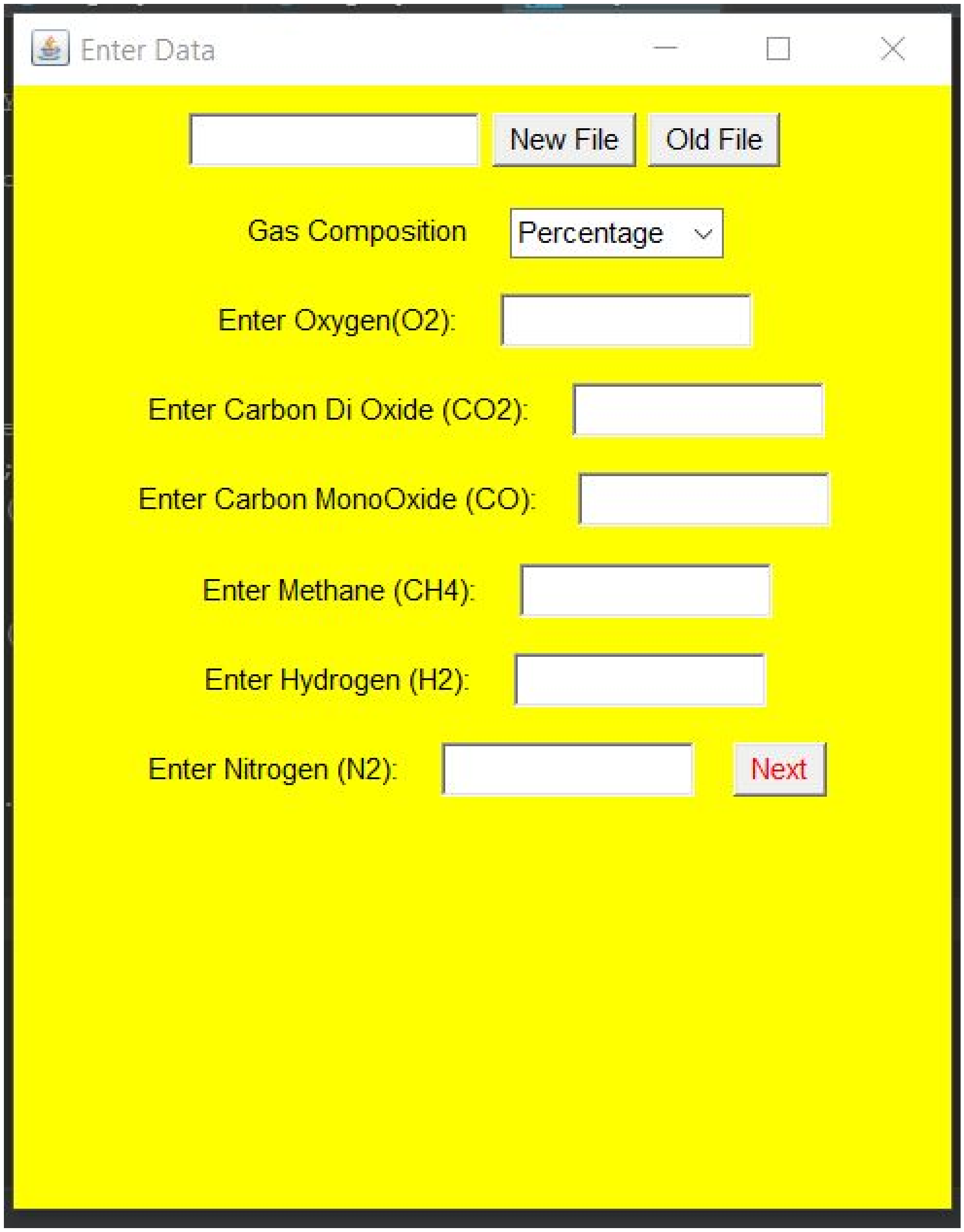
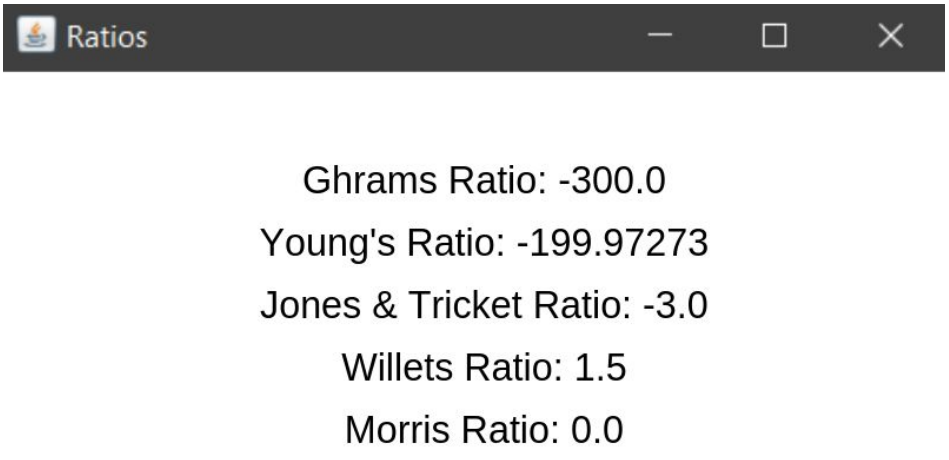


Fig. 2

Design


Enter Data
New File
Old File
Gas Composition
Percentage
Enter Oxygen(O2):
Enter Carbon Di Oxide (CO2):
Enter Carbon MonoOxide (CO):
Enter Methane (CH4):
Enter Hydrogen (H2):
Enter Nitrogen (N2):
Next


Ratios
Ghrams Ratio: -300.0
Young's Ratio: -199.97273
Jones & Tricket Ratio: -3.0
Willets Ratio: 1.5
Morris Ratio: 0.0

Coding

Class Module 1

```
public class FireDynamics {

   public static void main (String args[]) {

        Page1 pg1;
        pg1 = new Page1();

   }
}
```

Class Module 2

```
import javax.swing.*;
import java.awt.*;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;
import java.awt.event.WindowAdapter;
import java.awt.event.WindowEvent;

public class Page1{

   public Page1() {
       Frame f = new Frame("Fire Dyanamics");
       JLabel l1 = new JLabel("Fire Dynamics");
       l1.setFont(l1.getFont().deriveFont(50.0f));
       l1.setHorizontalAlignment(SwingConstants.CENTER);
       l1.setVerticalAlignment(SwingConstants.CENTER);

       Button GetStarted = new Button("Get Started");

       f.setBackground(Color.BLUE);
       l1.setSize(200,60);

       f.add(l1);
       f.add(GetStarted);
```

```
GetStarted.addActionListener(new ActionListener() {
    @Override
    public void actionPerformed(ActionEvent e) {

        Page2 p2 = new Page2("Enter Data");
    }
});

f.setLayout(new FlowLayout());
f.setSize(400,300);
f.setVisible(true);
f.addWindowListener(new WindowAdapter() {
    @Override
    public void windowClosing(WindowEvent e) {
        super.windowClosing(e);
    }
});

     }
}
```

Class Module 3

```
import javax.swing.*;
import java.awt.*;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;

import java.io.File;
import java.io.*;
import java.util.Scanner;

import static java.lang.System.*;



public class Page2 extends Frame {



   private static final Object OpenFilesEvent = null;
   String OldFileName;
   public Button next, b1, save;

   public float GhramRatio, YoungsRatio, JonesTrcketRatio;

   public File NewFile;

   public float i_carbonmono, i_oxygen, i_nitrogen, i_carbondi,
i_hydrogen, i_methane;
```

```
    TextField tf_newFile = new TextField(12);

    TextField tf_Oxygen = new TextField(10);
    TextField tf_CarbonDi = new TextField(10);
    TextField tf_CarbonMono = new TextField(10);
    TextField tf_Methane = new TextField(10);
    TextField tf_Hydrogen = new TextField(10);
    TextField tf_Nitrogen = new TextField(10);


    String columnn[] = {"Oxygen", "CarbonDi", "CarbonMono",
"Methane", "Hydrogen", "Nitrogen"};
    Label l_ghrams, l_youngs, l_jones, l_willets, l_morris;


    Page2(String title) {

        super(title);

        this.setVisible(true);
        setSize(400, 500);
        setResizable(true);
        setBackground(Color.yellow);
        setLocation(400, 100);
        out.println(this.getLocation());
        Label l = new Label("Gas Composition");


        Choice choice = new Choice();
        choice.add("Percentage");
```

```
        choice.add("Volume");

        //out.println(tfArray.length);


        Label l_Oxygen = new Label("Enter Oxygen(O2):");
        Label l_CarbonDi = new Label("Enter Carbon Di Oxide
(CO2):");
        Label l_CarbonMono = new Label("Enter Carbon MonoOxide
(CO):");
        Label l_Methane = new Label("Enter Methane (CH4):");
        Label l_Hydrogen = new Label("Enter Hydrogen (H2):");
        Label l_Nitrogen = new Label("Enter Nitrogen (N2):");
```

Search from old file

```
        Button OldFileCreate = new Button("Old File");


        OldFileCreate.addActionListener(new ActionListener() {
            @Override
            public void actionPerformed(ActionEvent e) {
```

Select Old File

```
                JFileChooser OldFileChoser = new JFileChooser();
                int response =
OldFileChoser.showOpenDialog((Component) OpenFilesEvent);
```

```
            if (response == JFileChooser.APPROVE_OPTION) ;
            {

                String OldFileName;
                tf_newFile.setText(OldFileName =
OldFileChoser.getSelectedFile().toString());

                out.println("File Selected");
                out.println(OldFileName);


                FileReader fr = null;
                try {
                    fr = new FileReader(OldFileName);
                } catch (FileNotFoundException e1) {
                    e1.printStackTrace();
                }

                BufferedReader br = new BufferedReader(fr);
```

Set Text to Text Field

```
                try {
                    String s;
                    int count = 0;
                    while ((s = br.readLine()) != null) {

                        byte[] bArray =
readFileToByteArray(OldFileName);
                        //displaying content of byte array
```

```
                    for (int i = 0; i < bArray.length;
i++) {
                        System.out.print((char)
bArray[i]);
                        if (equals(",")) {

                            count = count + 1;



                        }
                        System.out.println("Count " +
count);
                    }
                    String[] a = s.split(" ");

                    tf_Oxygen.setText(a[0]);
                    tf_CarbonDi.setText(a[1]);
                    tf_CarbonMono.setText(a[2]);
                    tf_Methane.setText(a[3]);
                    tf_Hydrogen.setText(a[4]);
                    tf_Nitrogen.setText(a[5]);

catch (IOException ex) {
                out.println("File Not Found");
            }

            try {
                fr.close();
            } catch (IOException e1) {
                e1.printStackTrace();
            }
```

```
            }
            Scanner OldFileName = new Scanner(in);
            OldFileName.next();

            File OldFile = new File("D:\\" +
String.valueOf(OldFileName));
            OldFile.getParentFile();

        }
    });
```

Create new file

```
    Button NewFileCreate = new Button("New File");
    NewFileCreate.addActionListener(new ActionListener() {
        @Override
        public void actionPerformed(ActionEvent e) {

            String NewFileName = tf_newFile.getText();

            NewFile = new File("D:\\S T U D Y\\P R O J E C T
S\\CSIR - CIMFR\\" + String.valueOf(NewFileName) + ".txt");
            try {
                NewFile.createNewFile();
            } catch (IOException e1) {
                e1.printStackTrace();
            }
            out.println("File Created Successfully");
        }
```

```
        });
```

Save into New File

```
        save = new Button("Save new file");
        save.addActionListener(new ActionListener() {
            @Override
            public void actionPerformed(ActionEvent e) {
                FileWriter fileWriter = null;
                try {
                    fileWriter = new FileWriter(NewFile);
                } catch (IOException e1) {
                    e1.printStackTrace();
                }
                BufferedWriter bufferedWriter = new
BufferedWriter(fileWriter);
                try {

                    bufferedWriter.write(tf_Oxygen.getText());
                    bufferedWriter.append(" ");
                    bufferedWriter.write(tf_CarbonDi.getText());
                    bufferedWriter.append(" ");
                    bufferedWriter.write(tf_CarbonMono.getText());
                    bufferedWriter.append(" ");
                    bufferedWriter.write(tf_Methane.getText());
                    bufferedWriter.append(" ");
                    bufferedWriter.write(tf_Hydrogen.getText());
                    bufferedWriter.append(" ");
                    bufferedWriter.write(tf_Nitrogen.getText());
                    bufferedWriter.write("\n");
```

```
        } catch (IOException e6) {
            e6.printStackTrace();
        }
        try {
            bufferedWriter.close();
        } catch (IOException e7) {
            e7.printStackTrace();
        }
        System.out.println("File Saved");
    }
});
```

Next Button

```
    next = new Button("Next1");
    next.addActionListener(new ActionListener() {
        @Override
        public void actionPerformed(ActionEvent e) {
            System.out.println("Next1 Clicked");
            String s_oxygen = (tf_Oxygen.getText());
            i_oxygen = Integer.parseInt(s_oxygen);

            if (i_oxygen > 60) {
                out.println("High Chance Oxygen");
            } else out.println("Low chance Oxygen");

            String s_carbondi = (tf_CarbonDi.getText());
            i_carbondi = Integer.parseInt(s_carbondi);
```

```
                    if (i_carbondi > 60) {
                        out.println("High Chance Carbon Di");
                    } else out.println("Low chance Carbon Di");

                    String s_carbonmono =
(tf_CarbonMono.getText());
                    i_carbonmono = Integer.parseInt(s_carbonmono);

                    if (i_carbonmono > 60) {
                        out.println("High Chance Carbon mono");
                    } else out.println("Low chance Carbon Mono");

                    String s_methane = (tf_Methane.getText());
                    i_methane = Integer.parseInt(s_methane);

                    if (i_methane > 60) {
                        out.println("High Chance methane");
                    } else out.println("Low chance methane");

                    String s_hydrogen = (tf_Hydrogen.getText());
                    i_hydrogen = Integer.parseInt(s_hydrogen);

                    if (i_hydrogen > 60) {
                        out.println("High Chance hydrogen");
                    } else out.println("Low chance hydrogen");



                    if (choice.getItem(choice.getSelectedIndex())
== "Percentage") {
                        out.println("You selected Percentage");
```

```
} else {
    out.println("You selected Volume");

}
out.println("\n successfully executed!!
Next");
```

Ratios

```
Frame f1 = new Frame("Ratios");
```

Graham's Ratio

```
float GhramRatio;
GhramRatio = (float) ((100 * i_carbonmono) /
((0.265 * i_nitrogen) - i_oxygen));
l_ghrams = new Label("Ghrams Ratio: " +
GhramRatio);

out.println("Ghrams Ratio " + GhramRatio);
```

Young's Ratio

```
float YoungsRatio;
YoungsRatio = (float) ((100 * (i_carbondi -
0.003)) / ((0.265 * i_nitrogen) - i_oxygen));
l_ghrams = new Label("Ghrams Ratio: " +
GhramRatio);

out.println("Young's Ratio " + YoungsRatio);
```

Jones & Tricket Ratio

```
                    float JonesTricketRatio;
                    JonesTricketRatio = (float) ((i_carbondi +
(0.75 * i_carbonmono) - (0.25 * i_hydrogen)) / ((0.265 *
i_nitrogen) - i_oxygen));
                    l_jones = new Label("Jones Tricket Ratio: " +
JonesTricketRatio);



                    out.println("Jones & Tricket Ratio " +
JonesTricketRatio);
```

Willet's Ratio

```
                    float WilletsRatio;
                    WilletsRatio = ((i_carbonmono) / (i_nitrogen +
i_carbondi));
                    l_willets = new Label("Ghrams Ratio: " +
WilletsRatio);
                    out.println("Willets Ratio " + WilletsRatio)
```

Morris Ratio

```
                    float MorrisRatio;
                    MorrisRatio = ((i_nitrogen) / (i_carbonmono +
i_carbondi));
                    l_morris = new Label("Ghrams Ratio: " +
MorrisRatio);



                    out.println("Morris Ratio " + MorrisRatio);
```

```
            f1.setForeground(Color.blue);
            f1.setLayout(new FlowLayout());
            f1.setSize(400, 300);
            f1.setVisible(true);
            f1.add(l_ghrams);
            f1.add(l_youngs);
            f1.add(l_jones);
            f1.add(l_willets);
            f1.add(l_morris);
        }
    });
```

Layout

```
next.setForeground(Color.red);

FlowLayout fl = new FlowLayout();

this.setLayout(fl);


Panel p01 = new Panel();
p01.add(tf_newFile);

p01.add(NewFileCreate);
p01.add(OldFileCreate);
```

```
// p01.add(tf_SelectFile);
this.add(p01);

Panel p0 = new Panel();
p0.add(l);
p0.add(choice);
this.add(p0);

Panel p1 = new Panel();
p1.add(l_Oxygen);
p1.add(tf_Oxygen);
this.add(p1);

Panel p2 = new Panel();
p2.add(l_CarbonDi);
p2.add(tf_CarbonDi);
this.add(p2);

Panel p3 = new Panel();
p3.add(l_CarbonMono);

Panel p6 = new Panel();
p6.add(save);

p6.add(next);
p3.add(tf_CarbonMono);
this.add(p3);

Panel p4 = new Panel();
p4.add(l_Methane);
```

```
        p4.add(tf_Methane);
        this.add(p4);

        Panel p5 = new Panel();

        p5.add(l_Hydrogen);
        p5.add(tf_Hydrogen);
        this.add(p5);

        Panel p7 = new Panel();
        p7.add(l_Nitrogen);
        p7.add(tf_Nitrogen);
        this.add(p7);

        this.add(p6);

    }

    private byte[] readFileToByteArray(String oldFileName) {
        return new byte[0];
    }
}
```

Testing

The various steps involved are explained below in a concise manner and an understanding of these would give a clear picture of the steps are to be followed.

Develop Test Cases

• Design test cases for the Unit Test event, based on the objectives of test and testing techniques to be used.

• Design test cases that include abnormal situations and input data. Design to expect the unexpected.

• Design test cases, testing for all known and expected error conditions.

Develop a program validation matrix

• Create a program validation matrix that identifies each test case and its requirements in order to track test case execution.

Identify test cycles

• Group test cases into manageable units {i.e.. cycles) that can be executed as a single run. Base the groupings on the relationship of objectives to each other and on the best way to quickly find the errors.

Develop test data descriptions

• Describe data that is available to use in the Unit Testing event. Descriptions must include the location of the data and any required special considerations for using it.

Develop test scripts

• Describe each step of the test procedure.

Test Plan contents as per Guidelines of IEEE STD 829-1998 (IEEE Standard for Software Test Documentation)

• Testing process

• Requirements traceability

• Tested items

• Testing Schedule

• Test recording procedures

• Hardware and software requirements

• Constraints

test design specifications as they are intended to be followed step-by-step and should not have extraneous detail.

Stages of Testing

The various stages involved in Testing are:

Unit Testing

This description is concerned with knowledge about testing a program unit, typically developed by a single individual, to determine that it is free of data, logic, or standards errors. This unit includes knowledge of dynamic analysis (equivalence partitioning, boundary value analysis, cause-effect graphing, logic-based testing, random testing, and syntax testing) and static analysis (complete path testing; decision testing, condition testing, and data-flow testing).

Integration Testing

This description is concerned with knowledge about validating that software components, which have been unit tested separately, interact correctly when they are put together to perform a higher order function. This unit also includes knowledge about dependency checking for calls, data, and processes, and about interface checking in terms of range, type compatibility, representation, number and order of parameters, and method of transfer.

System Testing

This description is concerned with knowledge about validating the specified functional requirements of a system. This unit includes knowledge about techniques to design and enact an independent testing process of all of the system's functions described in the software requirements specification.

Performance Testing

This description is concerned with knowledge about validating the performance requirements of a system. This unit includes knowledge about techniques to instrument

Performance measures like logging, event counts, event duration, and sampling. It also includes knowledge about methods for tuning a system for optimum saturation, load, and throughput threshold.

Acceptance Testing

This description is concerned with knowledge about validating the functional and non-functional requirements of a purchased or acquired system. This unit includes knowledge about techniques for using the contract, the statement of work, the software requirements specification, and the request for proposal to ensure that the delivered system meets all of the requirements (as perceived by the purchasing or acquiring organization).

Installation Testing

This description is concerned with knowledge about validating that a system will operate under all configuration possibilities. This unit includes knowledge techniques to perform configuration command checking in terms of rotation, and permutation of all physical, logical, and functional entities of a system.

Test Documentation

This description is concerned with knowledge about test plan preparation, test design specifications, test case specification, test procedure specification, test item transmittal reports, test log specification, test incident reports, and test summary reports.

Stages of Testing - Key Points

• Unit Testing- Individual components without any other system components

• Integration (Sub-System) Testing- Various components together. Not whole system.

• System Testing- Testing the entire system to find holes in linking together sub-systems.

• Acceptance Testing- Customer based test to make sure it meets all of the customer's requirements.

Testing Strategies

• **Top down** -starts with most abstract component and works downwards.

• **Bottom up** - starts with fundamental components and works upwards.

• **Thread** - used for multiple process systems where path leads through these various processes.

• Stress - overloading system to see if it can cope.

• **Back to back** - running the same tests on different versions to see if same output/ results is coming.

Major Types of Testing

- Functional Testing (Black Box).
- Structure Testing (White Box).
- Interface Testing
- Clean room Engineering and other novel concepts

USER MANUAL

Fire Estimation is a special purpose software designed to be used for CIMFR only. This software designed to be used by Mine Fire Scientists and related user.

The software starts with Get Started. We will have to click on the Get Started button to continue. Then the new window will open.

To Create New File

1. Enter File Name
2. Click New File
3. Enter Composition Of gasses
4. Click Save File
5. Click Next
6. You Will Get the Ratios

To Open New File

1. Click Old File
2. Select The File You Want To Open
3. It Will Fetch The Details
4. Click Next
5. You Will Get the Ratios

Why Java?

Today Java is the language of every programmer and you can also start learning Java for sure. Java is the best language that can be started with as it is very simple and easy to learn. Nobody in this world needs to have any prior knowledge of C and C++ programming language. Most importantly, you are going to find Java in most of the companies as the developers are interested to code in Java because of the relative stability of the language as it is platform independent. Java is used in both software and web development. Java is an Open Source Code programming language and therefore helps to give good insight into the language. Some of the advantages of learning Java are as follows:

• **Beginner friendliness:** It is an easy to learn programming language and beginner friendly as well.

• **Scalable:** It is easy to maintain as it is a statically typed language. This means your code will have to be checked for errors before it can be built into an app. Therefore, it becomes easier to track down the errors. In return, it becomes easier to maintain as your program grows in size and complexity.

• **Portability:** Java is highly portable language (the code written in Java can be moved from one computer system (platform) to another easily).

• **Bigger size Java community:** Java has 2nd largest StackOverflow community. Community size is important because the larger a programming language community is, the more support you'd be likely to get.

• **Java programmers are in demand:** Java language is the language of every programmer and thus, today, learning Java becomes the need of the hour.

Moreover, Android apps are also developed using Java since the Android Operating System runs on a Java language environment. So, learning Java opens up a broad way for development of android apps which are widely in demand these days.

Java is the language which is less challenging in the beginning but once you dive into the language you get to know that it is far more adventurous than any other programming language. Once you get a hold of the language, it will become easy all the way and when you master Java, you will out to be an outstanding and versatile developer.

Conclusion

Spontaneous heating and fire in coal mines is a major problem worldwide and has been a great concern both for the industry and researchers in this field. Majority of fires existing today in different coalfields are mainly due to spontaneous combustion of coal. The auto oxidation of coal ultimately leads to spontaneous combustion which is the major root cause for the disastrous of coal mine in leading and coal producing countries like USA, China, Australia, India and Germany.

It is a slow process and the heat evolved is carried away by air. This process of self-heating of coal or other carbonaceous material resulting eventually in its ignition is termed as “spontaneous heating” or “auto oxidation”. Coal can interact with oxygen in the air at ambient temperature liberating heat. If the heat is allowed to accumulate the interaction rate increases and may ultimately lead to fires – known as spontaneous fires.

So, to Detect the mine fire before it occurs by checking the composition of gas in the mines. I have tried to make this project which predicts about the fire in coal mines. So, we can prevent such a hazard to happen. To ensure the safety of the coal mine workers this project will help them by alarming before hand so that before the fire bursts out they can take preventive measures which will also save the lives of the workers thus ensuring more safety and security.