CHAPTER- 3

RESEARCH METHODOLOGY

The principle aim of this research concerns the outline of the elements participating in programming of simulation system. The interface library, database (oracle) library, and java libraries stored a set of usable components that aid the programmer to understand objective of software simulation development. The write particular code is a challenge for every requirement, and this code need to communicate with the applications introduced by the other programming libraries.

In this section the requirement configuration of the physics physical experiment is discussed. It starts with a diagram of a few architectures for speaking to physical science experimental instruments and the necessities for the outline of a programming transformation layer (PTL). This is initiated by a dialogue of some normal programming outline thoughts. As a physics scholar, we understood the problems of actual experimentation and as a solution to expensive, time consuming experiment process; we developed architecture for physics virtual experiment. At last, the configuration and usage of the programming transformation layer produced for this is presented.

3.1 Existing methodologies to build modeling of Physics Experiment Systems:

A transformation layer is a set of generalized models or calculations different from their particular physical execution. Supporting various physics experiment issues, the execution of experiment requires a theoretical framework construction modeling as a base for the virtual simulation. There have been numerous methodologies to build modeling of many science experiment systems (Sigmund, P., 1987; Perednis, D., 2005; Binder, K., & Heermann, D. W., 2010). An early methodology was the standalone framework (i.e. desktop applications), a facility for understanding basic physics experiments and their applications. Whilst fit for understanding a wide components involved in experiment, the framework worked on an extremely low level, generating a comparisons needed to upgrade many components of experimental system. This weakness with this sort of framework is extremely challenging to upgrade it to uphold collaborations with other complex frameworks, for example study of research activities concern with actual physics experiment.
Latest methodologies incorporate the Physics Simulation Software, which gives a measured schema for physics experiment trial. This dependent upon the "Object-Oriented" (Java) outline design. Objects communicate to physical segments, interface communicates to database as well as validation points. Also, no need of the physical experimental interface, as we stored pre-measured data for testing of simulated data against stored data. Previously many robotics simulations were developed by many researchers which needed physical connections between computer and experiment and in that case they used serial port programming which was one of disadvantage in terms of cost effective solution. The theoretical model approach taken by NASA gave an exceptional beginning stage for developing a dynamic representation of a test system programming, while the retargeting approach taken by other team empowered more particular characteristic. NASA's developed uniform unique interface dependent upon the connector design. The connector outline design built one interface class into an alternate uniform definition (Schneider, C., Porras, D., & Schaeetz, T., 2012). The utilization of this example empowered all simulations to be treated in a uniform way, but still physical connection between actual experiment and computer existed. With this background, we developed architecture to evaluate actual physics experiment and to transform into software interfaces.

Throughout the improvement of the programming for this theory, tasks developed for giving unique representations for virtual physics experiment. The use of virtual environment to study scientific concepts like the seismic wave, the earth’s coating cause of unsteadiness, sense of balance of radiation and also to study scientific process like control of telescope, movement of ocean, solar system (Arya, D. J., & Maul, A., 2012). Previous systems discussed before for virtual experimentation, which involved external interface is opposed in a transformation layer.

In 2011, Fabregas. al. presented idea of design of pedagogical agent for distance virtual experiment As per cross referral review, none of the past methodologies furnished an extensible plan to support various physics experiments. This could be ascribed to the way that a large portion of the past methodologies did not mean to back extremely numerous autonomous simulation solutions.

With a specific end goal to keep pace with the always showing signs of change simulation innovation and still research for component framework is needed
Components are a regular programming advancement method that permits a programming provision to give amplified practicality over the interest. To empower the programmer to build and collaborate with simulation without need of access to an actual experiment, the unique framework design might be consolidated with the component approach in the theoretical interface free development. This interface based configuration conveys the greater part of the experimental necessities:

- Easy access to database
- Quick communication with data over the internet
- Drag and drop facility for basic physics components like velocity, mass, viscosity calculations.

The utilization of the theoretical pluggable development empowers relative compile time for simulation execution. For the illustrative calculation to have the ability to unite and design a huge number of components, system takes more time for execution which in turn increases memory utilization of CPU (of computer) so it is beneficial to provide pre-developed components which can be added at the time of experiment testing. By advancing a set of standard interconnecting parts on top of the theoretical pluggable development framework is acknowledged where the data stream might be supervised and reconnected between programs, without considerations of the particular simulation environment. This empowers a more efficient way of experimental simulation execution.

3.1.1 Methodology for the developing a simulation model:

For the current research, following methodology for the developing a simulation model has been used. It consists of following important stages for developing and analysis of simulation real time model:-

A. Identify an existing system problem
B. Framing an existing system problem
C. Primary Data collection and detailing about variables existing system.
D. Preparation of plan diagram of simulation model and network diagrams of simulation model
E. Development and validation of real time based simulation model.
F. Back up data for future use
G. Experimental design selection for simulation model
H. Identifying experimental run conditions
I. Execution of condition of runs simulation model
J. Result recording and understanding the results
K. Recommendation for future work.

This list of steps follows a chronological order. However, several modifications and corrections will need to be made to the simulation study before we can fulfil the aims set out. It is not necessary to follow every single step, and extra steps might be needed to be carried out. We will now focus on these steps in the following three sections.

3.1.1.1 Development of simulation model:-

Each simulation model requires following quantity for realistic development of model i.e. detail about experimental unit, variables requires for input port, and there practical time relationship with the output. In chapter-4, in fist case of photoemission simulation model, the detail description of experiment is described and the change in intensity, frequency and voltage are the input variables, the total time require to perform the complete task = wait time + time require to perform is an practical time relationship with the output. Same type of components requires developing the second case of simulation model.

Development of simulation model for photoemission effect experiment and M-H curve are included following important stages:

A. Identify an existing system problem: -

The most important part of any simulation development system is problem of the existing system on which the simulation will work. Before the development of any model, the limitation and problem of actual system should know and hence the real experimental system of photoemission effect apparatus and M-H curve apparatus problems has been described in detail in
chapter-1 along with that need of simulation model is also described in same chapter.

B. Framing an existing system problem:-
All existing experimental units represent the some merits and some demerits. Before starting the work of development of simulation model, list the problem with existing unit and select any one problem for further study. After the selection of problem, the objective will define in detail for the present problem. Outline the interests and create hypotheses pertaining to the performance of the system. Finalize the duration of the study, for example, whether it will be a one-time decision making study or will it be used repeatedly over a period of time. Classify the target user, who will use the model. Ensure that the Problems are clearly illustrated.

For present study, the framing of the problem and objective of work regarding with real experiment are given in chapter 1.

C. Primary Data collection and detailing about variables existing system:
In this stage, the primary data will requires like different observation table and readings which will be collected from actual performance of experiment for the development of model. Along with primary data, the variables of existing experiments and time required for perform the activity has to known for model development. The primary data and input variables require for both simulation cases as stated above are describe in chapter 4. All this primary data are collected from actual experimental performance also verified and process under standard method of t probability methods for finding accurate results.

D. Preparation of flow chart of simulation model and network diagrams of simulation model:-
In this stage, research will develops the Software development life cycle (SDLC) theoretical model which will applicable to develop simulation model and this flow chart describes starting stage of that project to end of the project and application level. It is basically help to developer for set up the system assignment map. Software development life cycle (SDLC) having different
types as mentioned below. The detail Software Development Life Cycle (S.D.L.C.) and Execution part for photoemission effect apparatus and M-H curve apparatus model is described below in this chapter. Also the validation technique also described for same simulation models. The chapter 5 is output of this flowchart in which the developed simulation model is described in detail.

E. Development and validation of real time based simulation model: -

In this stage, well developed simulation model will be ready to test the output. The observation and results collected from the performance of simulation model will compare with the observation and results of existing experimental system. After this comparison the model will go through the statistically conclusion test and this test will make the comments on validation of simulation model. Development and validation process of simulation model for photoemission effect apparatus and M-H curve apparatus on the basis of their performance, their observation and output results are well described in chapter 5 of this thesis.

F. Back up data for future use:

The complete developed simulation model will available to the number of user and user do not have the information about the model hence there is need of documentation of objective of simulation model and input variables of simulation model for future use.

3.1.1.2 Process of simulation experiment Designing:

After the discussion of simulation model development methodology, the next important part is design process of simulation. It based on following stages and simulation experiment involves one or more tests wherein manipulations are performed on input variables of a simulation model so as to better understand the changes taking place in performance measures. The designing question of simulation model will resolve with discussion of each stage of simulation model designing. This designing part gives us information of a data we want like result of simulation model
of photoemission effect and B-H curve. Also it gives data as per our requirement. Following stages explains the Process of simulation experiment designing of photoemission effect and B-H curve.

G. Experimental design selection for simulation model:
For designing the simulation experimental model from actual or real experimental set up, we should find the input variable parameters and second is to find the factor whose make impact on the actual existing set up. If we found different input parameters then the system will be called as complex input variable hence in such system we will apply the documentation method. All the part of the system first will write in document and this documentation will be based on following criteria: a) Design Property Criteria, b) Advance Parameter or variable criteria, c) error optimization criteria.

   a) Design Property Criteria:
   In this type of criteria, first the developer will one or more property of the simulation for development the model and will fix the some property for simulation. Hence the final well developed simulation model will be strongly dependence on this property and its objective also fixed with its property. There are different criteria methods is available to set the property for design a simulation. The orthogonal property selection criterion is applicable for the first order polynomial type system. Some time system consist different points and different property but they are equal in number then orthogonal criteria failed to explain such system property hence in such case saturation design criteria is most applicable and usable but this approach is applicable upon some limited numbers of points.

   b) Advance Parameter or variable criteria:
   This criterion is completely different as compare to property criteria because in this criteria method the model design and documentation will be based on parameters. This criteria system allows to add or remove the parameters hence the design does not totally depends on property, it suggest another modification in design and it also satisfy some design property.

   c) Error optimization criteria:
When the system follows the property or parameter criteria for design purpose then there is chance of two type of error, one is bias error and another is variance errors. This can be controlled and minimized with error optimizing criteria.

H. Identifying experimental run conditions:

Simulate experiment designing require the condition for run, this conditions can collect from performing the experiment 3 to 5 times. The condition maybe two type, time dependant or time independent, some of the experiment run settings changes with time and hence output will be changes as per the change duration, but some of system setting does not changes even the time span of measurement too much large and the results value will be same as it is, for each measurement in any time. The condition of run changes with time then observation should take 3 to 5 instant. And compare each data with initial and set the correlation between them.

In case of photoemission model, the condition of runs was varies from one material to another cathode material and M-H curve model, condition was fixed and free from time.

I. Execution of condition of runs in simulation model:

According to stage G and H, the Experimental design selection criteria for photoemission effect and M-H curve has been discussed in next chapter also the condition for run have been listed. By using design criteria and condition for run, we have developed the simulation model and its Statistical inference is usually useful in evaluating the importance of an observed phenomenon. But some time Simulation data is frequently auto correlate and therefore by definition violate this assumption. The below steps are used to analyse data derived from simulations:

J. Result recording and understanding the results:

After the complete development of simulation model, the most important stage is result recording. The second last chapter of this thesis widely based on this stage. We developed the simulation model for photoemission and B-H curve apparatus and we had execute all the run condition and collected the
data from the simulation and represented in the form of table and graphs in same chapter. from this observation tables and graphs, the final conclusion is discussed in last chapter that is chapter of final thesis conclusion along with merit and limitation.

K. Recommendation for future work:
Finally at the end stage we have mentioned the future work with simulation can be possible. Also we have mentioned some suggestion regarding the simulation technique use in physics experimentations.

3.2 Software Development Life Cycle (S.D.L.C.) and Execution of It:-
Software development life cycle (SDLC) theoretical model which is used in simulation modeling for describing starting stage of that project to end of the project and application level. It is basically help to developer for set up the system assignment map.

3.3 SDLC Research Methodology and its various techniques significantly required for the current research:
In all over the world, Different websites and software developer are well familiar with software development life cycle. However there is one question related with the implementation of this type of planning. SDLC might assist to answer specific needs of different users. In some of cases, local developers develops software without any references of standard method but still the software is part of user group in this case there is no need of SDLC.

Following are the various SDLC methodologies are generally used in model developing process and ranges of SDLC are

A. Waterfall Software Development Life Cycle Model:
B. Prototyping Software Development Life Cycle Model
C. Iterative Enhancement Model
D. The Spiral Model
E. V- Model Methodology
F. Rapid Application Development (RAD)
G. Object Oriented Methodology
H. Joint Application Development (JAD)

By and large, two or more model can combine and develop the model which mentioned as hybrid model. In any type of model development there are two parallel stages one is fundamental documentation and second is flow chart. There are different types of models available for development of same type of project but the preference of model depends upon how the model is strictly close to the design of project.

3.3.1 System Development Life Cycle:

It consist different stages as shown in fig 3.1 which are described below:

Each and every system development life cycle of computer simulation model develops by using basically five stages namely:

1. SYATEM STUDY
2. POGRAMING STAGE
3. TESTING
4. IMPLEMENTATION
5. MAINTAINANCE
3.3.1.1 SYSTEM STUDY

In the system study, following different types of system study may be carried out

A) Study of Primarily System

B) Viability
C) Comprehensive

D) Analysis

E) Study Design of system

3.3.1.1 Primarily System Study:

This is also called system analysis stage in system study because it investigates basics requirement of designing system and also represents the pseudo image of physical system. It presents the system proposal those includes different points like description of statement of problem, objective behind the study which is going on, description of different definition and terms required for study, limitation of system, and description of future outcomes and advantages of system and so on. Lastly the whole first steps orients around the following points:

- Finding the objective of project,
- Introduction and commencement of project,
- Setting and Conditions scrutiny,
- Conclusion or findings regarding system proposal

3.3.1.2 Viability Study:

After finishing first step, the software analyst and strategy makers checks practicability of the system and perform the descriptive study on the proposed project. They decide different development steps of project and produce estimated budget of the project if the project is practically possible. The main aim behind descriptive study of project is search out the scope and to find the methods to solve the problem. In feasibility study, the software developer and strategy makers draws the exact amount of cost and benefits of the projects for the calculation of returns of investment.
### 3.3.1.3 Comprehensive System Study:

After checking practicability and possibilities of present project then it goes through detail system study in which the detail examination perform on the project regarding their objectives of projected project. Also there will be considered different points like different operation performing capability of the project, effect of these operations on project environment and in public also. These studies carry out on the basis of existing data files of current system, end results of current system and operation possible on present system.

By using positive and negative points of new system and also by considering the objective and new requirements of project, the frontier of project can be represented with exact manners.

The following points are the most essential part of this step:

- **Requirements:** - any system or models develops on based of the user requirement and user friendly concept by considering merits and limitation of user. For achieving user friendly nature it requires the specific arrangements and conditions.

- **Functional hierarchy:** It gives the detail stepladder of system will perform and relation between them.

- **Functional network:** the nature of functional network is similar to stepladder of system but it consist common description of task those very important as compare to the process.

- **Make an inventory of components of the main part:** Each and every entity described with the some features so before the study of any entity there is need of list of attributes.
3.3.1.1.4 Analysis of System:
In this step, system analyst analyses need of users, requirement of system, institute subject needs, changing process of subjects and accordance that he prepares a firm design for model. In another way analysis of system studies data which would be based on facts, be aware of the involve procedure in system, and also the study of suggestion which will help to improve the efficiency of system. This stage divides complex process in various subparts, offers the name to data storage and artificial process.

The main aim of any system analysis is to study the different question starts from its designing, working, nature, benefits, needs, uses, environment where it work and so on, the range and type of question depends upon resourcefulness and expertise of analyst.

3.3.1.1.5 System Design:
It is most important step in system development software, after performing the detailed studies on the analysis of system and user requirement then there is starting of designing the new system step is called system design.

The system design generally starts from the results of analysis of system and at the end it converts into physical model of virtual real world.

The design of simulation model consists of basically two stages namely fundamental design and Comprehensive design.

Basically all the system designs can be used following distinct paraphernalia and practices for designing the model:

- Plan of work
- Plan of collection of data
- Lexicon of records
- Smart English
- Preference spreadsheet
- Hierarchy of Assessment
The development and design of simulation model can be done with following steps:-

- By Exactly characterization of output of project
- By finding the data on which result of project depends
- By finding records and documents forms and setup
- By finding the procedure and software which is used for model development
- By finding the way of collection of results and way of entering records
- By planning of making codes
- By planning of handbook of process
- By creating documentation of processes

### 3.3.2 Programming Stage:

In this stage, developer starts the coding of design which is being passed through the above all the steps, and for making it workable on computer, it requires the coding process in where the system writes in computer machine understandable language is called programming language and this stage generally refer as programming stage.

The main objective of this stage is to write the complete program of design in language which is suitable, easy implementation and understand by computer. This is crucial stage because it decides the output, testing and maintenance of the system. The coding process effects on the cost of system because coding can increases or decreases efforts of testing and maintenance of system. Hence at the time of coding process, the objective of coding should be concentrate on simple programming of the project.

On another point of view this is very important stage because in which theoretically described model converts into the virtual real system with help of computer. Indirectly this stage makes its impact on maintenance of system and upcoming changes of the system.

### 3.3.3 Testing:

In this step, developers performs the checking on model after completing first to steps because the chances of errors are more possible in between the first two steps.
Testing process basically useful to test and verify computer models and its application make the acquaintance of technical necessities and its expected working.

The figure 3.2 shows the different basics steps of testing process. In below chart, the stages one to six are the primary carrying out before testing process and seven and eight phases are carrying out at and after testing process.

Figure 3.2- outlines the Test Approach

The model is a virtual computer system developed with machine coding program; it requires the testing for good and doing well system. The testing of model carries out on the base of whatever provided set of known results. The aim of the testing of
model is that to test match of output of model with expected output from the model. Commonly two type of test can be carried out on the model, namely:

a) **Test of Program:**

In this type of testing, the complete program codes goes through one by one testing process at the time of coding of program of computer simulation model, compiling program of computer simulation model and working situations of computer simulation model. If error is found then it pointed out and resolve with debugger.

b) **Test of complete model or project:**

After successful testing of programs and its codes and errors correction by debugging then testing method headed to test complete model which is run on actual information. After run of model, then it results of each actions recorded and that are compared with expected results if the output of model mismatched with expected output then the error identify in program and resolves it and again same way test carry out for expected results.

### 3.3.4 Execution, Implementation or Functioning:

After successful completion of above all the stages, the model will be available for user implementation or acceptance. In Execution stage, some of task can be performed by user like systematic arrangements of features of model and security of project, same time verifying the working status of feature of models, installation the model and finally get hold of model for run.

The most important actions in these steps are:

- Purchase of suitable required hardware and software for successful execution of model
- Setting up hardware and software in computer
- Adaptation
- Proper training to the user
Records and provide evidence

After successful implementation of the program on user computer then user training will begin. During the training of user following points are to be covered:

- To learn the method of execution of package
- To learn the procedure of input the data to model
- To learn the processing detail of model
- To learn the method of results and report generation from the model

After the complete training of user, then user capable to perform all the actions on computer based model or system.

3.4 Software Concepts:

For any software development framework an execution life cycle is needed for proficient improvement. To improve development that has the ability to furnish unique access to various software frameworks in a productive way a component based development was applied. Program coding is different from programming libraries, libraries are accumulation of subprograms which exhibit convenient capacities for the requisition programming and reusable code can be stored (like for database connection code, same code can be used again and again for different component development). Structures however furnish a set of interoperating dynamic (java) classes that gives a reusable plan for a particular sort of programming (Dietz, J. E., Chaturvedi, A. R., Black, D. R., Shaffner, J. E., Hsieh, C. H., Dunlop, S. & Burr, 2012). Component based programming characterizes an essential interchangeable framework empowering programming classes to be a solitary formed deployable unit (Rani, D., & Moreira, M. M., 2010).

An algorithmic has three key characteristics:

1. A database to store sample results of simulation results

2. Algorithm can automated for various hypothesis to test (Flowcharts)
3. Algorithm can use common user interface (homepage/screen) to calculate or display output.

These characteristics are main necessities for building web component based architecture. A programming facility offers a set of administrations for creating cases of different subclasses without expressly needing the name of the class we wish to construct (Fabregas, E., 2011). An interface facility broadens this by importing java packages to adopt the additional references without any alterations to the existing code itself. To implement java web interface/package the development never requires a (computer) registry that administers a rundown of all possible processes. When a “window based” component is made, the developer can look through the registry for the class, build it, and return it for utilization. But in our project we never need to worry about computer registry permissions from administrator as we developed architecture for web based component development which can be easily available to anyone any time over the internet. Previously, each class that should be receptive by means of the facility requires a system that permits a duplicate of code to be made, and in addition a strategy to include its data to the computer registry. By making a static duplicate of the class, the data is immediately enlisted at the exact start of the requisition, after any client code is executed. Applications of this approach for C++ are given by C++ dummies (book).

The previous essential unique interface processing configuration was amplified in this postulation with novel augmentations permitting the development of classes from Dynamic Link Libraries (dll) and the class initiation data into a differentiate structure to permit forming of items. But, in case of java web development we never need dll as it deals with C++ programming. Instead of that java used packages which contain readymade classes which can be evaluated easily.
The Unified Modeling Language (UML) diagram for our simulation framework is shown in Figure 3.3, and the application execution pseudo code for this approach is given below.

1. Open home page of physics online simulation
2. Select experiment from list
3. Set parameters for experiment
4. Run experiment

5. Test results with database result

6. Store test results with Test ID.

Figure 3.4 – Sequence diagram

Figure 3.4 shows a sequence diagram for the sample experiment. This type of diagram provides step by step execution of activities. First (User Interface) UI is accessed by user to login to our web tool. After login, user can select experiment and
timer gets started. Now user can view experimental simulation area where he/she can perform experiment with ready web components. If results are satisfactory, data gets stored into database.

3.4.1 Web Based Design:

The online simulation framework is unique, extensible as well as with reusable classes. This will provide interoperable facilities. This system is web based hence, anyone can work with this system same time. This correspondences system is executed on top of the conceptual framework, by giving a dynamic uniform interface that empower the associations between various web components. The ideal model implemented to realize this is an information stream developed with physics as a interdisciplinary development. Every java class that is a part of the dataflow framework characterizes a set of data and yield channels. Classes which transform inputs are characterized as channels.

The JSP flowchart shown in figure 3.5, this chart shows flow of processes that all participates in physics simulation web tool. It likewise gives techniques for supervising the diagram: including, uprooting, emulating, and process associations. The java classes themselves keep up a record with their stored info and yield positions. The main class is likewise solution for supervising information stream throughout execution. By the following flowchart, every module could be visualized clearly and joined together with any input to system. The parameters needed for the input can either be manually stipulated or immediately created from run time database information extraction. Flow sequence can likewise conduct additional sequence of code execution to computerize simulation of the physics experimental framework. Coupling the pluggable java interface with a sequence diagram gives an adaptable, configurable and extensible programming stage. The runtime database environment needed for this framework as physical interfacing of physics experimental inputs/outputs restricted. On the other hand, there are numerous more operations presupposed throughout the introductory set up. This extra set up time is significant in examination as compare to other simulation architectures.
3.5 Physics Experimental Architecture:

The Physics Experimental Architecture is the programming developed for this simulation theory that empowers different physics experiments to be executed from given input data. It provides uniform modifying interface to different rapid development frameworks to the programmer. This architecture methodologically spins around the dynamic pluggable interface. There are various outline objectives for the simulation architecture:
• Common architectural platform: Provide a simple user interface and architectural simulation for physics experiments

• Development Flexibility: Allow extra characteristics to be joined in a dynamic impromptu support

• Platform Independent: Provide open source software language support and permitting interoperation with multi-domain frameworks.

• Completeness of physics simulation: Expose however much purpose as could be expected for every basic physics experiments.

• Common Component Implementation: Identical purpose empowered on different experiments needs collected and utilized together

• Scalability: Provide a common interface, that could be utilized similarly well on any computer server.

3.5.1 Component Orientation and Utilization:

The simulation with which COU are built is the thing that empowers COU to have a flexible, extendable and maintainable structures. For every structure in COU, there is a dynamic class representation that is furnished as the interface to the programmer. The concrete class execution inherits from the theoretical class and additionally gives a static method strategy which is utilized to enroll the application presence with the COU development environment. The development supports a testing of every module usage and in addition the relating simulation framework. The point when the programmer starts the development of any module/ component, the main java class identifies the class relationships involving in simulation framework and builds the best suited sequence of instructions.

The COU administers many tasks for every class. This also forms different test cases which further utilized to evaluate particular usage variant. In case of such testing event that we wish to make common for all modules we require a simulation execution for each part of experiment. The java execution might inherit from the JDK
package object, and furnish usefulness to make a replica of itself, as well as the stored data. This incorporates the adaptation of the usage, the name of the class (e.g. "Spring"), and the usage bunch it fits in with (e.g. "damping"). An elective physics simulation usage might give the same data, aside from a diverse section for the whole experiment (e.g. "overload"). The point when a programmer shows the output they wish to utilize (e.g. "mass"), the program runs its request from Tomcat 6.x server to incorporate the most relevant data for the given experiment and Tomcat Server fetches data from database. Further this data feed to algorithm for calculations and then output will be displayed on main screen.

The point when the programmer makes a request to Tomcat Server, the application works with request and responses generated by user, this flow of activities shown in Further, user can enter “0http://localhost:8080/” in web browser to start application server i.e. Tomcat Server prior to start physics experimental home page.

3.5.2 Object Representation And Mathematical Modeling:

The physics itself is a mathematical based science which requires a geometrical conversion to represent its objects like shape, velocity, mass etc. To support most extreme similarity, repetitive executions of comparable instructions are given. For instance, not all interfaces give a relation between a physical form, and a basic Object. Accordingly, inside COU there are two approaches to represent an object. Once, as a COU inflexible form that holds predefined Object, and once as a COU as joined rigid body/sphere representation. Along with these, if an experiment developed for the purpose for sequential requirements, then when a COU is applied, it will be difficult to append an Object to actual simulation.

3.5.2.1 Simulation of Rigid Body:

There are various notable cases to think about for the representation of rigid bodies to study basic application input in physics experiment. The process to simulate the geometrical shapes (here sphere as a rigid bodyFigure 3.6) is as follows:
• Draw sphere and mark radius, circumference etc.
• Study geometrical parameters of the sphere.
• Write formula for each parameter.
• Ex: Assume origin point “O” and radius “r”
  i.e. OX=OY=OZ=r
  Hence,
  Area of circle given by,

\[ \pi r^2 \]

• Create problem definition, design problem and write pseudocode to calculate area of circle like follows:

Programmers write Algorithms in terms of pseudocode which are the informal and artificial language, practically simple and uncomplicated hence pseudocode are called tool of text based algorithm design. All the programs statements shows shuffle up enslavement, it consist of while loop, do, for, if else loop, switch loop so on. The following examples are illustration of this concept. (http://web.uetaxonomy.edu.pk )

![Figure 3.6–Sphere Geometry](image-url)
Problem Definition

You require a program that prompts the user for the length and width of a rectangle and the radius of a circle. The program should calculate and display the rectangle perimeter and rectangle area and the circumference of the circle. Write a defining diagram and pseudocode for this problem.

Defining Diagram

<table>
<thead>
<tr>
<th>Input</th>
<th>Processing</th>
<th>Output</th>
</tr>
</thead>
<tbody>
<tr>
<td>length</td>
<td>Prompt and Read length, width, radius</td>
<td>perimeter area</td>
</tr>
<tr>
<td>width</td>
<td>Calculate perimeter</td>
<td></td>
</tr>
<tr>
<td>radius</td>
<td>Calculate area</td>
<td>circumference</td>
</tr>
<tr>
<td></td>
<td>Calculate circumference</td>
<td></td>
</tr>
</tbody>
</table>

Generalized Algorithm (pseudocode)

Method Main

Prompt for length, width, radius

Read length, width, radius

perimeter = 2*length + 2*width

area = length * width

display perimeter and area

circumference = 2*Pi*r
display circumference

End

**Generalized Program Solution**

Algorithm with separate methods to calculate perimeter, area and circumference

Public static void main{

Prompt for length, width, radius

Read length, width, radius

perimeter = Call Calc_Perim(length, width)

area = Call Calc_Area(length, width)

display perimeter and area

circumference = Call Calc_Circ(radius)

display circumference  }

End

//Following method will give perimeter of circle

Calc_Perim (l, w){

return l * w; }

End

//Following method will calculate Area of circle

Calc_Area (l, w){

A = 2 * l + 2 * w;

return A
Calc_Circ(r) {
    C = 2*Pi*r
    return C
}

Pseudocode for calculation of volume of Cylinder

Using problem statement and design steps mentioned before, we can calculate volume of cylinder. The pseudocode is as follows which we can convert to programmatically structure later.

//declare your variables const
float pi = 3.14159; double
radius, height, volume;
// I/O
out 'enter your radius'
in radius;
out 'enter your height'
in height
//calculation
The volume of a cylinder equals the (area of the base)*height = \Pi r^2 h
//so
pi * pow(radius,2) * height
//print result
out volume;

From above process for representation of geometrical parameters in software simulation it is very clear that we can convert or transform anything which is physically available to simulation architecture. The main inputs need to know is
“parameters of experiment”. We divided physics experiment into small logical parts to write java code for each part which we call “modules”.

Further we developed java classes in directory of Tomcat Server which researcher can use at the time of research experiment. The benefits of this system are as follows as per our study:

1. Researcher can study each module of experiment

2. We can modify any input parameter to test experiment and its effects.

3. We can reduce human errors which happen in case of physical experiment.

4. We can store our findings in database.

5. If parameter is unsupported; however different geometries could be utilized to solve the problem of this sort. In the event that the object is totally new, it is completely executed in java class structure.

In the other case, the Object is not supported by the physics, yet just considers it as a static body (2D) first and then make replica to consider it as a rigid body. This is a normal case for numerous objects, for example planes, maps etc. Since it is a normal case, representation for the geometries is given as "baseline" representations. The baseline can consider maps, polygon, and planes. A divide refinement is made between hub adjusted planes, and planes which can have any introduction, so as to give most extreme uphold for simulations which don't furnish an orientated plane representation.

Provided that a simulation can just communicate to a plane object as a static body, then it is absolutely considers to it as a landscape. This helps abstain from confounding the client with having object accessible that can't be relegated to planes. Overall, the object representation is accessible, however can just be utilized for making static forms. Assuming that the client endeavors to appoint an object to a dynamic figure then system will give a failure. The other case is taken care of in a comparative way. The Object has an executed representation inside simulation system, however returns a failure when the Object is utilized with a dynamic form. Further case can be where the Object will just uphold system alerts, yet won't back...
static body. Also the case can be like Object is unsupported straightforwardly by the
physics on the other hand, it might be considered to utilizing existing methods. To
handle this system utilizes the arrangement provided at design stage i.e. to make small
modules initially. Current architecture furnishes a conceptual set of systems for
making an object oriented representation as a set of geometry, empowering the
simulation tool to build the more straightforward object utilizing models.

The other class is geometries that are not subsets of an upheld Object, yet could be
assessed from the structure of an elective. A case of this is if a simulation does not
furnish a plane representation, be that as it may furnish uphold for a static body
rather. This is taken care of with total and extraordinary case study. This may not
bring about the Object being underpinned all around by the simulation, rather
similarly as a hard-coded framework representation. Since not many simulations help
powerful figures with curved Object, an elective rigid body is furnished that can hold
different less complex geometries. This empowers complex Object to be recreated
from compound basic geometries. Most simulations help a compound form sort; on
the other hand, not many simulations uphold progressively added or uprooting
geometries to a body once they are simulated. This represents an issue since a few
simulations require the formation of a base and afterward connecting geometries to it.
In many cases the geometries require to be characterized when the base is considered.

This is accomplished by utilizing design assessment. Every Object is added to the
design is archived in a support and once the body is concluded all the geometries
could be transformed inside one call, paying little heed to how the underlying
simulation requires the compound design to be developed. There are various extra
differentiate rigid bodies that may be backed by a simulation.

• Static structure – Bodies that are not moveable, for example landscape. These exist
only for impact location.

• Kinematic structure – Bodies that could be moved, yet don't react to constrains or
impacts. These permit questions move in predefined movements and communicate
with the dynamic protests in the scene.

• Dynamic structure –Bodies that react to drives and crashes. The development of
these figures is ascertained from the powers connected to the form.
• Compound structure– Bodies that hold various geometrical parameters.

If a static or kinematic structure object is replicated with a dynamic design the all constrains following up on the object might be discredited. That is, if gravity is, no doubt connected to object, energy might be connected of the same size in the inverse bearing in which gravity is acting. Over the long period numerical distinctions between the set position of the design and the position endeavoring to be upheld begin separating and the figure will start to float.

A few simulations give incomplete usage of a design. Simulations will regularly furnish the capacity to reposition static object bodies, empowering a portion of the practicality of a kinematic object to be attained.

On the other hand, this development does not impact any design that lie on the way between the static object beginning position and completion position. Accordingly the static object might cover or completely hold a dynamic form, prompting precarious results. To evade these insecurities this purpose is not considered in present study.

3.6 Transformation of Objects:

To make the dynamic system, the Object is ordinarily constrained to straightforward shapes, for example basic geometrical shapes. The Object shapes upheld by the simulation underlying simulation framework are not expressly demonstrated previously. A system does give a multi-dimensional representation of the geometries so as to apply simulations that just support convex dynamic objects.

Squares are represented as a set of 8 planes, whereas the sphere and cylinder are generated from a specified number of divisions

3.6.1 Experimental Object Representations: Environmental Effect:

To build reasonable situations for a test system, various components are presupposed, incorporating systems for consideration of environmental factors. For example, the properties of fluids and effect of environment.
For instance, when spray particles considered for physics experiment for spray pyrolysis, a material science investigation in a research center may be possible physically but to simulate it, we need to simulate its molecular structure too.

3.7 Prototype Representations:

The main difference between simple physics experiment and physics experiment with some fluid interaction is consideration of special material properties like liquid or semi liquid, powdered form of substance or substance which changes properties with temperature etc.

As we focusing over spray pyrolysis experimentation as a case study and it’s feasibility for simulation, we need to take overview of such properties of material and how can we simulate it in physics simulation tool is important.

Many tools available in market like “Blender Fluid Simulation” but those commonly deal with only fluid simulation and not with geometric simulations. We developed architecture for spray pyrolysis case study with such combined properties which may prove better in future.

3.8 Fluid Simulation Program:-

We developed a realistic liquid simulation using java applet.

Outline of development:

We stored current and previous states of liquid in two maps. The image or screen area is with same size of the each map, here we also applied a single array cater for all states; also, we added two rows for each state.

- Each frame will toggle between stored value maps and current reading map.
  
  Use of offset to change location of initial state into the array.

- In the present state array of the each array elements ,the following points are needed:
• Check number of pixel of left side, number of the pixel on right side, number of pixel on below and above. Count it and take the addition of it and divide by two so that it can work well on right side.

• After the addition, subtract the map of present state.

• For the smooth pattern, the intensity of each uneven distribution should be peter out in each pass and keep it away from build up from the surface. Practically it can be possible by reducing its fraction of heights. After performing above action, there would need of optimization with right shift way. Here is right shift taking place with five and intensity of uneven distribution reduced 1/32nd of previous time of each right shift action.

• The next step is to deform a background image in the present state due to the height of uneven distribution of water wave and this can be archiving by the value of offset. The offset(X/Y offset) can calculated on the base of simple formula that is present distance of from the center of wave record and wave magnitude at same point.

• At the end, offset set at the positive and it should be less than image size of texture.

In this chapter we overviewed the properties of solid geometrical objects and its way of software simulation implementation. Also we developed sample pseudocode for the physics simulation for sphere. As we may need fluid properties while study of spray pyrolysis case study in subsequent chapter, we provided information along with java applet simulation program for simulation of water (fluid).