BATCH - 15 - LATEX CODE

```
%preamble section
\documentclass[12pt, a4paper]{report}
\usepackage[utf8]{inputenc}
\usepackage{graphicx}
\usepackage{array}
\title{BATCH 15 - SE LAB - RECORD - SOFT COPY}
\author{20VV1A12-(59, 60, 61, 63)}
\date{\today}
\usepackage{pgf}
\usepackage{pgfpages}

\pgfpagesdeclarelayout{boxed}
{
  \edef\pgfpageoptionborder{0pt}
}
{
  \pgfpagesphysicalpageoptions
  {%
    logical pages=1,%
  }
  \pgfpageslogicalpageoptions{1}
  {
    border code=\pgfsetlinewidth{2pt}\pgfstroke,%
    border shrink=\pgfpageoptionborder,%
    resized width=.95\pgfphysicalwidth,%
    resized height=.95\pgfphysicalheight,%
    center=\pgfpoint{.5\pgfphysicalwidth}{.5\pgfphysicalheight}%
  }%
}

\pgfpagesuselayout{boxed}
%preamble section completed
% start of body
\begin{document}

\maketitle
\title\begin{center}
   \underline{ \textbf{SRS DOCUMENT  }}
\end{center}
\textbf{Section 1 : Introduction}
\newline
\textbf{Purpose : }
\begin{itemize}
      \item This document describes the software requirements for a Ariane 5 Rocket Launching System. It is intended for the designer, developer, the maintainer and the user of the system.
\end{itemize}
\textbf{Scope : }
\begin{itemize}
      \item The Rocket Launching System is designed to build a distributed software application that manages the launching of Ariane 5 rocket.
      \item Illustrate issues with requirements specification, multi organisational working, artificial system validation and some of the problems of software reuse.
      \item It also shows the organisational complexity of systems of development and how organisational issues can lead to system failure.
\end{itemize}
```

```
    \textbf{Definitions/Abbreviations : }
    \begin{itemize}
        \item Ballistic : moving under the force of gravity.
        \item Terminal : forming or situated at the end or extremity of something.
        \item Mach : The ratio of the speed of a body to the speed of sound in the
surrounding medium.
        \item Altitude : The height of an object or point in relation to sea level or ground level.
        \item Flight : The action or process of flying through the air.
    \end{itemize}
    \textbf{References : }
    \begin{itemize}
        \item https://iansommerville.com/software-engineering-book/case-studies/ariane5/
        \item http://www.rvs.uni-bielefeld.de/publications/Incidents/DOCS/Research/Rvs/
Misc/Additional/Reports/ariane.html
        \item www.google.com
    \end{itemize}
    \newpage

    \textbf{Section 2 : General Description }
    \newline
    \newline
    \textbf{Overall Description : }
    \newline
    This section usually consists of six subsections, as follows :
    \begin{itemize}
        \item Product Perspective
        \item Product Functions
        \item User Characteristics
        \item Constraints
        \item Assumptions and Dependancies
        \item Requirements
    \end{itemize}
    \textbf{Product Perspective : }
    \newline
    The product is a software designed to launch a Rocket. Constitutes Internal Calculators
that help in maintaining the rocket in regular parameters of flight.
    Scientific, Development and Maintenance Team interacts with the software.\
    \newline
    \newline
    \textbf{Product Functionality : }
    \newline
    Product provides calculators like Ballistic Flight, Terminal Velocity, Rocket Altitude, Mach
and Speed of Sound. Each has its own importance in launching a rocket.
    \newline
    \newline
    \textbf{Design and Implementation Constraints : }
    \begin{itemize}
        \item As the product is inherited from the case study of Ariane 5 Launch.Accident, the
rocket has blasts in 37 secs into the flight after launch.
        \item Reason being no proper code reuse.
        \item The content didn't provided enough information about how we can develop a
complete end to end rocket launching software.
        \item Also many companies keep their source code confidential making it much more
difficult to get ideas related to it.
        \item Understanding how various items are involved and how they can be managed
into one so that they can lead to meaningful outcomes it's a long task.
    \end{itemize}
    \newpage
    \textbf{Assumptions and Dependencies : }
    \begin{itemize}
```

```
            \item Any of the modern operating systems are supported.
            \item Windows, Linux, Max are basic OS that are fully supportive.
            \item Browser should be of the latest generation.
            \item Browsers like Internet Explorer are not supported.
            \item An active internet connection is highly recommended.
            \item No further packages or software plugins are required.
            \item The specific hardware and software due to which the product will be run.
        \end{itemize}
        \newpage
        \textbf{Section 3: Specific Requirements}
        \newline\newline
        \textbf{External Interface Requirements:}
        \newline\newline
        \textbf{Functional Requirements:}
        \newline\newline
        \textbf{User Interfaces:}
        \begin{enumerate}
          \item \textbf{Home Page}: \begin{itemize}
             \item For Providing navigation buttons to different features in the software.
             \item This home page is designed for showing the features of the software
designed.
             \item contains BFC, TVC, RAC, MSSC
             \item User will input numeric values and output is shown.
          \end{itemize}
          \item \textbf{Ballastic Flight Calculator}
          \begin{itemize}
             \item This program calculates the maximum height of a launched ballastic shell or a
shell with drag, and the time from launch when the maximum height is reached.
             \item has attributes Initial Velocity, Mass, Cross Section Area etc.
             \item User will input numeric values and output is shown. Further user can use RAC
and home navigation buttons.
          \end{itemize}

          \item \textbf{Terminal Velocity Calculator}
          \begin{itemize}
             \item This Program Calculates the Terminal Velocity of a falling object. Circular Orbit
Calculator and velocity of an object in a circular orbit about he earth, moon, or mars.
             \item has mass, Cross Section Area, Drag Coefficient etc.
             \item User will input numeric values and output is shown. Further user can use RAC
and home navigation buttons.
          \end{itemize}
          \item \textbf{Rocket Altitude Calculator}
         \begin{itemize}
            \item This program calculates the terminal velocity of a falling object.
            \item Angle A, Angle B, Angle C etc.
            \item User will input numeric values and output is shown. Further user can use RAC
and home navigation buttons.
         \end{itemize}
        \end{enumerate}
        \item \tetxbf{Mach and Speed of Sound Calculator}
        \begin{itemize}
          \item This program calculate the speed of sound as a function of temperature and the
mach number as fucntion of object speed and speed of sound.
          \item has attributes Altitude, Speed etc.
          \item User will input numeric values and output is shown. Further user can use RAC
and home navigation buttons.
        \end{itemize}
        \newpage
        \textbf{Non Functional Requirements}
        \newline\newline
```

```
	\textbf{Performance Requiremenrs}
	\begin{enumerate}
	\item The Rocekt Altitude Calculator (RAC) will calculate the Altitude of the rocket when needed by other interfaces.
	\item The Terminal Velocity of Rocket when needed by other interfaces.
	\item All calculators should work in coordination with each other.
	\end{enumerate}
	\newline\newline
	\textbf{Safety and Security Requirements:}
	\begin{enumerate}
	    \item It's advised to ensure the right usernames and password for authentication of scientist or Launch Engineer.
	    \item Product may not allowed to use by general users
	    \item Systems running the software must satisfy the hardware and feature requirements for obtaining the maximum accurate and less error prone output from the product.
	    \item Systems running the software should maintain antivirus and firewall protection for reliable performance of the product.
	\end{enumerate}

	\newline
	\textbf{Software Quality Attributes:}
	\begin{itemize}
	    \item Portability
	    \item Interoperability
	    \item Modifiability
	    \item Performance
	    \item Availability
	    \item Re-usability
	    \item Scalability
	    \item verifiability
	    \item Manage Complexity
	    \item Security
	    \item Openness
	\end{itemize}


	\newpage\begin{center}
	\title{\textbf{\underline{USE CASE DIAGRAM}}}
	\end{center}
	\textbf{Description:}
	\begin{center}
	    \begin{tabular}{| m{5cm} | m{5cm}| }
	    \hline
	        Name: & Ariane 5 Rocket Launch Calculator\\
	    \hline

	        Actors: & Launch Engineer, Test Engineer and Software Development Engineer.\\

	    \hline
	        Triggers: & User opens application and initiates software.\\
	    \hline
	        Pre-Condition: & 1. The Hardware components must be in good working condition.\\
	    \hline
	    & 2. The Calculators must take in accurate input and accurate output.\\
	    \hline
	        Post-Condition: & 1. All performed operations and their results must be stored.\\
	        \hline
	    & 2. Repeated Testing must be conducted to identify errors.\\
	    \hline
```

```
            \textbf{Actor Actions} & \textbf{System Actions}\\
            \hline
            1)The Launch Engineer launches the rocket. & 3)Request sent to Terminal Velocity
Calculator to provide terminal velocity value
          \\
            \hline
        2)Launch Engineer provides input to the Ballistic Flight Calculator & 5)Request sent to
Rocket Altitude Calculator to provide altitude value
        \\
        \hline
        4) Launch Engineer provides input to the Terminal Velocity Calculator & 6) Final output
given by Ballistic Flight Calculator
        \\
        \hline
        7)Launch Engineer provides input to the Mach and Speed of Sound Calculator & 8)
Output given by Mach and Speed of Sound Calculator\\
        \hline
            \end{tabular}
        \end{center}
        \begin{center}
            \begin{tabular}{| m{5cm} | m{5cm}| }
            \hline
        \textbf{Alternative Flow of Events:} & 1)In case errors are encountered by the software,
the Test Engineer refers to the Log Report and reports the errors to the Software Development
Engineer to fix the errors.\\
        \hline
        & 2)User must ensure that they have entered the correct values. If they have not they
must re enter the values to get the correct \\
        \hline
          \end{tabular}
        \end{center}
        \newpage
        \begin{figure}
            \centering
            \includegraphics[width=\linewidth]{usecase.jpeg}
            \caption{}
            \label{fig:my_label}
        \end{figure}
        \newpage
        \begin{center}
            Usecase Diagram for Ariane 5
        \end{center}
        \newpage
        \begin{center}
            Usecase Diagram code:
        \end{center}
        @startuml



        package "Development Team"{



        actor "Software Development Engineer" as SDE



        }
```

```
package "Maintenance Team"{


actor "Test Engineer" as TE


}



package "Scientific Team"{


actor "Launch Engineer" as LE


}
rectangle "Rocket Launch Calculator"{
usecase "Ballastic Launch Calculator" as BFC
usecase "Terminal Velocity Calculator" as TVC
usecase "Rocket Altitude Calculator" as RAC
usecase "Mach and Speed of Sound Calculator" as MSSC
usecase "Log Record" as LR
}

BFC .down $>$ TVC : $<<$ include $>>$

RAC .up $>$ BFC : $<<$ extend  $>>$

RAC ..$>$ TVC : $<<$ extend $>>$

RAC ..$>$ MSSC : $<<$ extend $>>$


BFC --down--$>$ LR : $<<$ include $>>$
```

```
TVC .down$>$ LR : $<<$ include $>>$

RAC .down$>$ LR : $<<$ include $>>$

MSSC .down$>$ LR : $<<$ include $>>$


TVC --| right-|$>$ BFC

MSSC --right-|$>$ BFC




LE --$>$ BFC

LE --$>$ MSSC




SDE -right-$>$ TVC

SDE -right-$>$ BFC

SDE -right-$>$ RAC

SDE -right-$>$ MSSC




TE -left-$>$ LR

@enduml
\newpage
\title\begin{center}
   \underline{\textbf{{Class Diagram  }}}
\end{center}

  \textbf{Aim:  To draw class diagram for notes and file management}

  \textbf{Description:}
```

```
            \begin{itemize}
              \item Class Diagram : In software engineering,a class diagram in the Unified
Modelling Language(UML) is a type of static structure diagram that describes the structure of a
system by showing the system classes ,their attributes,operations and the rectangle among
objects.
              \item Class : A description of a group of objects with similar roles in the
system.Simply user can say that class is a combination of data numbers and member functions.
              \item  Class Notation : A Class notation consists of three parts.
              \begin{enumerate}
                 \item Class Name :
                 \begin{itemize}
                    \item The name pf the class appears in the first position
                 \end{itemize}
                 \item Class Attribute :
                 \begin{itemize}
                    \item Attributes are shown in the second partition .
                    \item The attribute type is shown after the colon.Attributes map on to member
variables in code.
                 \end{itemize}
                 \item Class Operations(Methods) :
                 \begin{itemize}
                    \item Operations are shown in the third partition .
                    \item The return type of a method shown after the colon at the end of the
method signature.
                \end{itemize}
              \end{enumerate}


            \end{itemize}
            \textbf{Description of class:}
          \begin{enumerate}
             \item Ballastic Flight Calculator(BFC):
             \begin{itemize}
                \item BFC is the class name and it is one of the key actors of the rocket launch
system.
                \item Initial velocity,mass,cross section area,altitude,drag coefficient,terminal
velocity,max height are attributes.
                \item Compute BFC() is for computing ballastic flight.
                \item home(),calcu;ate rocket altitude() are supporting ,methods.
             \end{itemize}
           \newpage
            \item Terminal Velocity Calculator(TVC):
            \begin{itemize}
               \item TVC inherits attributes methods, from BFC.
               \item Compute TVC(), is method for calculating terminal velocity.
            \end{itemize}
            \item Rocket Altitude Calculator(RAC):
            \begin{itemize}
               \item Rocket Altitude Calculator is a supporting class and also a autonomous class in
the system.
               \item angle A,angle B, angle C ,angle D , reference length.
               \item Compute RAC() is method for calculating RAC.
            \end{itemize}
           \item Mach and Speed of Sound Calculator(MSSC):
           \begin{itemize}
              \item Speed , Speed of sound ,mach are attributes.
              \item inherits BFC class
              \item Compute MSSC() is individual method.
           \end{itemize}
           \item Log Record(LR):
```

```
       \begin{itemize}
          \item Log Report class is used to store data of all.
          \item Signal is an attribute.
          \item Calculate BFC(),TVC(),RAC(),MSSC() all methods of the class.
       \end{itemize}
      \end{enumerate}
      \textbf{Relationship used in the class diagram as:}
      \begin{enumerate}
         \item Inheritance:A solid line with a hollow arrow head that point from the child to the
parent class.
         \item Simple Association : A line connectivity between two classes.
         \item Composition : A special type of aggregation which denotes strong ownership
between two classes when one class is part of another class.
         \item Aggregation : Aggregation relationship shows a class as if it's a part of or
subordinate to another class free.
      \end{enumerate}
      \begin{figure}
         \centering
         \includegraphics[width=\linewidth]{class.jpeg}
         \caption{Class Diagram}
         \label{fig:my_label}
      \end{figure}
      \newpage
      \begin{center}
         Class Diagram for Ariane 5
      \end{center}
      \newpage
      \textbf{Java code for Ballastic Flight Calculator}
      \newline

        import java.util.Vector;

      public class BFC

      \{

        private float initialVelocity;

        public float mass;

        public float crossSectionArea;

        public float altitude;

        public float dragCoefficient;

        public float terminalVelocity;

        public float maxHeight;

        public float timeToMaxHeight;

          public LR myLR;

          public Vector  myTVC;

          public TVC myTVC;

          public TVC myTVC;
```

```
     public RAC myRAC;


    public float computeBFC()

    \{

    return 0.0;


    \}

    public void home()
    \{

    \}

    public float calculateRocketAltitude()
    \{

    return 0.0;

    \}

\}
\newpage
\textbf{Java code for Terminal Velocity Calculator}
\newline
import java.util.Vector;

public class TVC extends BFC
\{



     public Vector  myBFC;

     public BFC myBFC;

     public Vector  myRAC;

     public BFC myBFC;



    public float computeTVC()

    \{

    return 0.0;


    \}


\}
\newpage
\textbf{Java code for Rocket Altitude Calculator
}
\newline
```

```
import java.util.Vector;


public class RAC

\{


  private float angleA;


  private float angleB;


  private float angleC;


  private float angleD;


  private float referenceLength;


  public float altitude;


    public LR myLR;


    public Vector  myTVC;

    public MSSC myMSSC;

    public MSSC myMSSC;

    public Vector  1;


  public float computeRAC()

  \{

  return 0.0;

  \}
```

```
\}
\newpage
\textbf{Java code for Mach and speed of sound calculator}
\newline

import java.util.Vector;


public class MSSC extends BFC
\{


  private float speed;


  public float speedOfSound;


  public float mach;


    public Vector  myLR;

    public RAC myRAC;

    public RAC myRAC;


  public float computeMSS()

  \{

  return 0.0;

  \}


\}
\newpage
\textbf{Java code for Log Report
}
\newline

import java.util.Vector;


public class LR

\{
  private boolean signal;



    public BFC myBFC;

    public Vector  myMSSC;

    public RAC myRAC;


  public float calculateBallasticFlight()

  \{

  return 0.0;

  \}


  public float calculateTerminaVelocity()

  \{

  return 0.0;


  \}


  public float calculateRocketAltitude()
  \{

  return 0.0;

  \}


  public float calculateMSoS() \{

  return 0.0;

  \}


\}

\newpage

\begin{center}
    \underline{\textbf{Sequence Diagram}}
```

```
        \end{center}
         \textbf{Aim :  To draw sequence diagram for rocket launch software}
         \newline\newline
         \textbf{Description :}
         \begin{itemize}
           \item Launch Engineer inputs data
           \item BFC receives input and computes ballastic flight
           \item TVC receives input and computes terminal velocity
           \item RAC receives input and computes rocket altitude
           \item MSSC receives input and computes Mach and Speed of sound
           \item Test Engineer request for log data
           \item Log record gives response back to Test
         \end{itemize}

          \textbf{Description of roles :}
          \begin{itemize}
              \item Launch Engineer : Responsible for activating software requires valid credentials
to access the system input data for derived calculator.
              Receives response back from the system.
              \item Test Engineer : Responsible for identifying bugs and errors in the system.
              \item BFC : Receives input ,computes ballastic flight , sends response back to
launch engineer.
              \item TVC : Receives input from user , compute terminal velocity and sends
response back to user.
              \item RAC : Takes input from launch engineer, computes Rocket Altitude Calculator
and sends response back.
              \item MSSC : Takes input,calculates mach and speed of sound.
              \item LR : Records every transaction in the system.

          \end{itemize}
         \newpage

           \textbf{Sequence Diagram Code : }

           @startuml

           actor "Launch Engineer" as foo


           actor "Test engineer" as foo1

        participant "BFC" as foo2

        participant "TVC" as foo3

        participant "RAC" as foo4

        participant "MSSC" as foo5



        database "LR" as foo6


        activate foo6

        $foo1 \rightarrow $foo6 : request Log Report


        $foo6 \rightarrow $foo1 : response of Log Sheet
```

```
$foo \rightarrow $foo2 :inputs data
alt successful case
activate foo2
$foo2 \rightarrow $foo : response about Ballastic flight
else some failure
$foo2 \rightarrow $foo : input proper values

$foo2 \rightarrow $foo6 : save data into log

deactivate foo2
end


$foo \rightarrow $foo3 : inputs data
alt successful case
activate foo3
$foo3 \rightarrow $foo : response about Terminal velocity
else some failure

$foo3 \rightarrow $foo : input proper values
$foo3 \rightarrow $foo6 : save data into log
deactivate foo3
end
$foo \rightarrow $foo4 : inputs data
alt successful case
activate foo4

$foo4 \rightarrow $foo : response about Rocket Altitude
else some failure
$foo4 \rightarrow $foo : input proper values
$foo4 \rightarrow $foo6 : save data into log
deactivate foo4
```

```
end

$foo \rightarrow $foo5 : inputs data

alt successful case

activate foo5

$foo5 \rightarrow $foo : response about speed and mach

else some failure

$foo5 \rightarrow $foo : input proper values

$foo5 \rightarrow $foo6 : save data into log

deactivate foo5

deactivate foo6

end

@enduml

\begin{figure}
   \centering
   \includegraphics[width=\linewidth]{sequence.jpeg}
   \caption{Sequence diagram}
   \label{fig:my_label}
\end{figure}
\newpage
\textbf{Ballastic Flight Calculator Code:}

@startuml

actor "Launch Engineer" as foo

actor "Test engineer" as foo1

participant "BFC" as foo2

participant "TVC" as foo3

participant "RAC" as foo4

participant "MSSC" as foo5



database "LR" as foo6

activate foo6

$foo1 \rightarrow $foo6 : request Log Report



$foo6 \rightarrow $foo1 : response of Log Sheet
```

```
$foo \rightarrow $foo2 :inputs data
alt successful case
activate foo2
$foo2 \rightarrow $foo : response about Ballastic flight
else some failure
$foo2 \rightarrow $foo : input proper values
$foo2 \rightarrow $foo6 : save data into log
deactivate foo2
end

$foo \rightarrow $foo3 : inputs data
alt successful case
activate foo3
$foo3 \rightarrow $foo : response about Terminal velocity
else some failure
$foo3 \rightarrow $foo : input proper values
$foo3 \rightarrow $foo6 : save data into log
deactivate foo3
end

$foo \rightarrow $foo4 : inputs data
alt successful case
activate foo4
$foo4 \rightarrow $foo : response about Rocket Altitude
else some failure
$foo4 \rightarrow $foo : input proper values
$foo4 \rightarrow $foo6 : save data into log
```

```
deactivate foo4

end



$foo \rightarrow $foo5 : inputs data

alt successful case

activate foo5

$foo5 \rightarrow $foo : response about speed and mach

else some failure

$foo5 \rightarrow $foo : input proper values

$foo5 \rightarrow $foo6 : save data into log

deactivate foo5

deactivate foo6

end

@enduml
\newpage
\begin{figure}
    \centering
    \includegraphics[width=\linewidth]{s1.jpeg}
    \caption{Sequence diagram 1}
    \label{fig:my_label}
\end{figure}
\newpage
\begin{center}
    Ballastic Flight Calculator for Ariane 5
\end{center}
\newpage
\textbf{Terminal Velocity Calculator Code:}

@startuml

actor "Launch Engineer" as foo

actor "Test engineer" as foo1

participant "TVC" as foo3



database "LR" as foo6

activate foo6

$foo1 \rightarrow $foo6 : request Log Report



$foo6 \rightarrow $foo1 : response of Log Sheet
```

```
$foo \rightarrow $foo3 : inputs data

alt successful case

activate foo3

$foo3 \rightarrow $foo : response about Terminal velocity

else some failure


$foo3 \rightarrow $foo : input proper values

$foo3 \rightarrow $foo6 : save data into log

deactivate foo3

end



@enduml
\newpage
\begin{figure}
   \centering
   \includegraphics[width=\linewidth]{s2.jpeg}
   \caption{Seqential diagram 2}
   \label{fig:my_label}
\end{figure}
\newpage
\begin{center}
   Terminal Velocity Calculator for Ariane 5
\end{center}

\newpage

\textbf{Rocket Altitude  Calculator Code:}

@startuml

actor "Launch Engineer" as foo

actor "Test engineer" as foo1

participant "RAC" as foo4



database "LR" as foo6

activate foo6

$foo1 \rightarrow $foo6 : request Log Report



$foo6 \rightarrow $foo1 : response of Log Sheet
```

```
$foo \rightarrow $foo4 : inputs data

alt successful case

activate foo4


$foo4 \rightarrow $foo : response about Rocket Altitude

else some failure

$foo4 \rightarrow $foo : input proper values

$foo4 \rightarrow $foo6 : save data into log

deactivate foo4

end



@enduml
\newpage
\begin{figure}
    \centering
    \includegraphics[width=\linewidth]{s3.jpeg}
    \caption{Sequence Diagram 3}
    \label{fig:my_label}
\end{figure}
\newpage
\begin{center}
    Rocket Altitude Calculator for Ariane 5
\end{center}
\newpage
\textbf{Mach and Speed of Sound Calculator Code:}


@startuml



actor "Launch Engineer" as foo

actor "Test engineer" as foo1

participant "MSSC" as foo5



database "LR" as foo6

activate foo6

$foo1 \rightarrow $foo6 : request Log Report
```

```
$foo6 \rightarrow $foo1 : response of Log Sheet



$foo \rightarrow $foo5 : inputs data

alt successful case

activate foo5

$foo5 \rightarrow $foo : response about speed and mach

else some failure

$foo5 \rightarrow $foo : input proper values

$foo5 \rightarrow $foo6 : save data into log

deactivate foo5

deactivate foo6

end

@enduml

\newpage
\begin{figure}
    \centering
    \includegraphics[width=\linewidth]{s4.jpeg}
    \caption{Sequence diagram 4 }
    \label{fig:my_label}
\end{figure}
\newpage
\begin{center}
    Mach and Speed of Sound Calculator
\end{center}
\newpage

\textbf{Log:}

@startuml

actor "Test engineer" as foo1

database "LR" as foo6

activate foo6

$foo1 \rightarrow $foo6 : request Log Report



$foo6 \rightarrow $foo1 : response of Log Sheet

@enduml
\newpage
\begin{figure}
    \centering
    \includegraphics[width=\linewidth]{s5.jpeg}
```

```
        \caption{Sequential diagram 5}
        \label{fig:my_label}
    \end{figure}
    \newpage
    \begin{center}
        Log for Ariane 5
    \end{center}
    \newpage
    \begin{center}
        \textbf{Collaboration diagram}
    \end{center}


     \textbf{Aim:-To draw a collaboration diagram for rocket for rocket launch   software}
     \newline


     \textbf{Description}
     \begin{itemize}
         \item The collaboration diagram used to show the relationships between the object
and systems
         \item Instead of showing how the message ,It despects the architecture of recording
in the system as it based on object oriented programming
         \item The collaboration diagram which is also known as communication diagram is
used portray objects architecture in the system
         \item the diagrams use the sequence diagram and collaboration diagram to define
and clarify the roles of objects that performing particular parts of events of use case
     \end{itemize}

     \newpage
    \begin{center}
        \textbf{Collaboration Diagram Code}
    \end{center}

    @startuml

    actor "Launch Engineer" as foo

    actor "Test engineer" as foo1

    participant "BFC" as foo2

    participant "TVC" as foo3

    participant "RAC" as foo4

    participant "MSSC" as foo5

    autonumber


    database "LR" as foo6

    activate foo6

    $foo1 \rightarrow $foo6 : request Log Report



    $foo6 \rightarrow $foo1 : response of Log Sheet
```

```
$foo \rightarrow $foo2 :inputs data

alt successful case

activate foo2

$foo2 \rightarrow $foo : response about Ballastic flight

else some failure

$foo2 \rightarrow$foo : input proper values

$foo2 \rightarrow$foo6 : save data into log

deactivate foo2

end


$foo \rightarrow $foo3 : inputs data

alt successful case

activate foo3

$foo3 \rightarrow $foo : response about Terminal velocity

else some failure

$foo3 \rightarrow $foo : input proper values

$foo3 \rightarrow $foo6 : save data into log

deactivate foo3

end


$foo \rightarrow $foo4 : inputs data

alt successful case

activate foo4

$foo4 \rightarrow $foo : response about Rocket Altitude

else some failure

$foo4 \rightarrow $foo : input proper values

$foo4 \rightarrow $foo6 : save data into log

deactivate foo4
```

```
        end



        $foo \rightarrow $foo5 : inputs data

        alt successful case

        activate foo5

        $foo5 \rightarrow $foo : response about speed and mach

        else some failure

        $foo5 \rightarrow $foo : input proper values

        $foo5 \rightarrow $foo6 : save data into log

        deactivate foo5

        deactivate foo6

        end

        @enduml
        \newline
        \newpage
         \begin{center}
            \textbf{Collaboration diagram}
         \end{center}
        \begin{figure}
             \includegraphics[height=120mm, width=120mm]{collab.jpeg}
             \caption{Collaboration diagram for Ariane 5}
             \label{fig:my_label}
         \end{figure}
        \newpage
        \begin{center}
            \underline{\textbf{COCOMO MODEL}}
        \end{center}
        \textbf{AIM:-}
        To estimate the cost of the personal insulin pump in a cocomo model
        \newline
        \newline
        \textbf{DESCRIPTION:-}
        \newline
        Cocomo (Constructive Cost Model) is a regression model based on LOC, i.e number of
Lines of Code. It is a procedural cost estimate model for software projects and is often used as a
process of reliably predicting the various parameters associated with making a project such as
size, effort, cost, time, and quality. It was proposed by Barry Boehm in 1981 and is based on the
study of 63 projects, which makes it one of the best-documented models.

        The key parameters which define the quality of any software products, which are also an
outcome of the Cocomo are primarily Effort & Schedule:
        \newline
        \newline
        \textbf{Effort:}
         \begin{itemize}
            \item
        Amount of labor that will be required to complete a task. It is measured in person-months
units.\\
        \end{itemize}
        \newline
        \textbf{Schedule:}
        \begin{itemize}
          \item
        Simply means the amount of time required for the completion of the job, which is, of
course, proportional to the effort put in. It is measured in the units of time such as weeks, months.
        \item Different models of Cocomo have been proposed to predict the cost estimation at
different levels, based on the amount of accuracy and correctness required. All of these models
can be applied to a variety of projects, whose characteristics determine the value of constant to
be used in subsequent calculations. These characteristics pertaining to different system types are
mentioned below.
        \end{itemize}
        \newline
        \textbf{Boehm’s definition of organic, semidetached, and embedded systems:}
        \newline
        \newline
        \textbf{Organic }–
        \newline
        \begin{itemize}
        \item
        A software project is said to be an organic type if the team size required is adequately
small, the problem is well understood and has been solved in the past and also the team
members have a nominal experience regarding the problem.
        \end{itemize}
        \newline\newline
        \textbf{Semi-detached }–
        \newline \newline
        \begin{itemize}
          \item
        A software project is said to be a Semi-detached type if the vital characteristics such as
team size, experience, knowledge of the various programming environment lie in between that of
organic and Embedded. The projects classified as Semi-Detached are comparatively less familiar
and difficult to develop compared to the organic ones and require more experience and better
guidance and creativity. Eg: Compilers or different Embedded Systems can be considered of
Semi-Detached type.
        \end{itemize}

        \textbf{Embedded} –
        \newline \begin{itemize}
          \item
        A software project requiring the highest level of complexity, creativity, and experience
requirement fall under this category. Such software requires a larger team size than the other two
models and also the developers need to be sufficiently experienced and creative to develop such
complex models.\\
        All the above system types utilize different values of the constants used in Effort
Calculations.\\
        \end{itemize}
        \newline
        \textbf{Types of Models:}



        \newline
        \newline
        COCOMO consists of a hierarchy of three increasingly detailed and accurate forms. Any
of the three forms can be adopted according to our requirements. These are types of COCOMO
model:\\
        \newline
```

```
            \begin{itemize}
            \item  Basic COCOMO Model
            \item Intermediate COCOMO Model
            \item Detailed COCOMO Model
            \end{itemize}
            The first level, Basic COCOMO can be used for quick and slightly rough calculations of
Software Costs. Its accuracy is somewhat restricted due to the absence of sufficient factor
considerations.\\

            Intermediate COCOMO takes these Cost Drivers into account and Detailed COCOMO
additionally accounts for the influence of individual project phases, i.e in case of Detailed it
accounts for both these cost drivers and also calculations are performed phase-wise henceforth
producing a more accurate result. These two models are further discussed below.
            \newline
            \textbf{Estimation of Effort: Calculations }–
            \newline \newline
            \textbf{Basic Model –}
            \newline
              E= $ a(KLOC)^$ b
              \newline
             time= $ c(Effort)^$ d
             Person required = Effort/ time \\
            The above formula is used for the cost estimation of for the basic COCOMO model,\\ and
also is used in the subsequent models. The constant values a,b,c and d\\ for the Basic Model for
the different categories of system:\\
            \begin{center}
            \includegraphics[scale=1.0]{1.jpeg}\\
            \end{center}
            The effort is measured in Person-Months and as evident from the formula is dependent
on Kilo-Lines of code.\\
            The development time is measured in months.\\

            These formulas are used as such in the Basic Model calculations, as not much
consideration of different factors such as reliability, expertise is taken into account, henceforth the
estimate is rough.\\

            \textbf{Intermediate Model –}
            \newline
            The basic Cocomo model assumes that the effort is only a function of the number of lines
of code and some constants evaluated according to the different software systems. However, in
reality, no system’s effort and schedule can be solely calculated on the basis of Lines of Code. For
that, various other factors such as reliability, experience, Capability. These factors are known as
Cost Drivers and the Intermediate Model utilizes 15 such drivers for cost estimation.\\
            \raggedright
            \textbf{Classification of Cost Drivers and their attributes:}\\
            \newline
            \textbf{(i) Product attributes –}\\

            Required software reliability extent\\
            Size of the application database\\
            The complexity of the product\\

            \textbf{(ii) Hardware attributes –}\\

            Run-time performance constraints\\
            Memory constraints\\
            The volatility of the virtual machine environment\\
            Required turnabout time\\
            \textbf{(iii) Personnel attributes –}\\
            \newline
```

```latex
	Analyst capability\\
	Software engineering capability\\
	Applications experience\\
	Virtual machine experience\\
	Programming language experience\\
	\textbf{(iv) Project attributes –}\\
	\newline

	Use of software tools\\
	Application of software engineering methods\\
	Required development schedule\\
	\begin{center}
	\includegraphics[width=\linewidth]{1.jpeg}\\

	\end{center}
	\textbf{Detailed Model –}\\
	\newline
	Detailed COCOMO incorporates all characteristics of the intermediate version with an assessment of the cost driver’s impact on each step of the software engineering process. The detailed model uses different effort multipliers for each cost driver attribute. In detailed cocomo, the whole software is divided into different modules and then we apply COCOMO in different modules to estimate effort and then sum the effort.\\
	\textbf{The Six phases of detailed COCOMO are:}\\
	\n

	Planning and requirements\\
	System design\\
	Detailed design\\
	Module code and test\\
	Integration and test\\
	Cost Constructive model\\
	The effort is calculated as a function of program size and a set of cost drivers are given according to each phase of the software lifecycle.\\

	\begin{center}
	\includegraphics[width=150mm,height=150mm]{cocomo-1.jpeg}\\
	\end{center}
	\begin{center}
	\includegraphics[width=150mm,height=150mm]{cocomo-2.jpeg}\\
	\end{center}
	\newpage

	\begin{center}
	\underline{\textbf{Function Point Analysis}
	}\end{center}
	\newline
	\textbf{Aim: Calculate effort using FP oriented estimation model}
	\newline\newline
	\textbf{Description:}
	\newline\newline
	FPA provides standardized method to functionally are the software work product. This work product is the output of software new developmeist and improvement projects for subsequent releases is the software which is relocated to the production application at project implementation. It measures functionality from the users' point of view on the bass of what the user requests and receives in return.
	\newline
	Function Point Analysis (FPA) is a method or set of rules of Functional Size Measurement R assesses the functionality delivered to as unen, based on the user's external view of the functional requirements. It measures the logical view of an application not the physically
```

```
implemented view or the internal technical view.The Function Point Analysis technique is used to
analyses the functionality delivered by software and Unadjusted Function Point (UFP) is the unit of
measurement.
          \newline\newline
          \textbf{Objectives of FPA:}
          \newline
          \begin{enumerate}
             \item The objective of FPA is to measure functionality that the user requests and
receives.
             \item It should be simple enough to minimize the overhead of the measurement
process.It should be a consistent measure among various projects and organizations
          \end{enumerate}
          \newline
          \textbf{Types of FPA:}
          \newline
          \begin{enumerate}
             \item Transactional Functional Type
             \begin{itemize}
                \item External Input (EI): EI processes data or control information that comes from
outside the application's boundary.
               \item External Output (EO): EO is an elementary process that generates data or
control information sent outside the application's boundary.
               \item External inquiries (EQ): EQ is an elementary process made up of an input-
output combination that results in data retrieval.
          \end{itemize}
          \newpage
             \item Data Functional Type
             \begin{itemize}
                \item  Internal Logical File ((LF): A user identifiable group of logically related data or
control information maintained within the boundary of the application.
                \item External Interface Fae (EIF): A group of user recognizable logically related data
allusion to the software but maintained within the boundary of another software.
             \end{itemize}
          \end{enumerate}
          \newline
          \begin{center}
             \textbf{Calculation of Function Point Analysis for Ariane 5 Rocket Launch Software}
          \end{center}
          \begin{enumerate}
             \item External Input (EI)
             \begin{itemize}
          \item Initial Velocity
          \item Mass
          \item Cross section area
          \item Drag coefficient
          \item Altitude
          \item Angle A
          \item Angle B
          \item Angle C
          \item Angle D
          \item Reference length
          \item Speed
            \end{itemize}
          \item External Output (EO):
          \begin{itemize}
          \item Terminal velocity
          \item Max height
          \item Time to Max height
          \item Terminal velocity
          \item Altitude
```

```
        \item Speed
        \item Speed of sound
        \item Mach
        \end{itemize}
        \item External Inquiries (EQ)
        \begin{itemize}
        \item Home page
        \item Ballastic flight calculator
        \item Terminal velocity calculator
        \item Rocket altitude calculator
        \item Speed of Mach of sound calculator
        \end{itemize}
        \item  Internal Logical File (ILF):
        \begin{itemize}
        \item Control Access
        \item File into
        \item File type
        \item Backup maintainence
        \item Input/Output log record
        \end{itemize}
        \item External Interface File (IFS)
        \begin{itemize}
          \item User
          \item User management
          \item System admin
          \item Testing admin
          \item Launch admin

        \end{itemize}

        \end{enumerate}

        \newpage
        \textbf{Counting Function Point (FP):
        }
        \newline
        Step-1:  $F=14 \times  $scale
        \newline
        Scale varies from 0 to 5 according to character of Complexity Adjustment Factor (CAF)
Below table shows scale.
        \newline
        \begin{enumerate}
        \item No Influence
        \item Incidental
        \item Moderatte
        \item Average
        \item Significant
        \item Essential
        \end{enumerate}
        \newline
        On the Average Case we Estimate Scale as Average Scale=3
        \newline
        $F=14 \times $3
        \newline
        F=42
        \newline
        Step 2: Calculate Complexity Adjustment Factor (CAF)


        CAF= $0.65\times $(0.01)
        CAF=$0.65+ $($0.01\times$42)

        CAF= $ 0.65+ $0.42

        CAF=1.07

        \newline
        Step-3: Calculate Unadjusted Function Point (UFP). Multiply each individual function
point to corresponding values in TABLE


        \begin{figure}
          \centering
          \includegraphics[width=\linewidth]{Functional point calculator.jpg}
          \caption{FPC}
          \label{fig:my_label}
        \end{figure}


        \newpage
        As weighting factors are also average (even in question) hence we will multiply each
individual function point to corresponding values in TABLE


        \newline
        User Input=32



        User Output=19



        User Inquiries=5



        User Files=3



        External Files=1



        UFP= (32*6)+(19*7) +(5*6)+(3*15)+(1*10)
        UFP=192+133+30+45+10
        UFP=410

        Step 4: Calculate Function Point.

        FP=UFP * CAF

        FP=410*1.19

        FP=487.9
        \newpage
        \begin{figure}
```

```
            \centering
            \includegraphics[width=\linewidth]{fp.jpg}
            \caption{Function point}
            \label{fig:my_label}
        \end{figure}
        \newpage
        \begin{center}
            \textbf{Function point Counter}
        \end{center}



        \newpage
        \begin{center}
            \underline{\textbf{GANTT CHART}}
        \end{center}
        \textbf{AIM:-}
        To develop a Timeline chart and project table using PERT or CPM project scheduling
methods.
        \newline
        \newline
        \textbf{DESCRIPTION:-}
        \newline
        \newline
        A Gantt chart is a graphical depiction of a project schedule. It's is a type of bar chart that
shows the start and finish dates of several elements of a project that include resources,
milestones, tasks, and dependencies. Henry Gantt, an American mechanical engineer, designed
the Gantt chart.
        \newline
        \newline
        \textbf{How Gantt Charts Work}
        \newline
        \newline
        The Gantt chart is the most widely used chart in project management. These charts are
useful in planning a project and defining the sequence of tasks that require completion. In most
instances, the chart is displayed as a horizontal bar chart.
        \newline
        \newline
        Horizontal bars of different lengths represent the project timeline, which can include task
sequences, duration, and the start and end dates for each task. The horizontal bar also shows
how much of a task requires completion.
        \newline
        \newline
        [Important: The length of the bar is proportional to the time necessary for a task's
completion. The project tasks are represented on the vertical axis.]
        \newline
        \newline
        A Gantt chart helps in scheduling, managing, and monitoring specific tasks and
resources in a project. The chart shows the project timeline, which includes scheduled and
completed work over a period. The Gantt chart project manager in communicating project status
or plans and also helps ensure the project remains on track.
        \newline
        \newline
        \textbf{Benefits of a Gantt Chart}
        \newline
        \newline
        The chart identifies tasks that may be executed in parallel and those that cannot be
started or finished until other tasks are complete. The Gantt chart can help detect potentialbootle
necks and identify tasks that may have been excluded from the time line.
        \newline
```

```latex
        \newline
        The chart depicts task slack time or additional time for completion of a task that should
not delay the project, noncritical activities that may be delayed and critical activities that must be
executed on time.
        \newline
        \newline
        Gantt charts can be used in managing projects of all sizes and types. These charts are
utilized in several industries and for a range f projects, such as building dams, bridges and
highways, software development, and development of other goods and services. Project
management tools, such as Microsoft Visio, Project, SharePoint, andexcel or specialized software,
such as Gantto or Matchware, can help in designing Gantt charts.
        \newline
        \newline
        \textbf{Historical development}
        \newline
        \newline
        Although now regarded as a common charting technique, Gantt charts were considered
revolutionary when first introduced. The first known tool of this type was developed in 1896 by
Karol Adamiecki, who called it a harmonogram. Adamiecki did not publish his chart until 1931,
however, and only in Polish, which limited both its adoption and recognition of his
authorship.construction project. It appears that Schürch's charts were not notable but rather
routine in Germany at the time they were published The prior development leading to Schürch's
work is unknown. Unlike later Gantt charts, Schürch's charts did not display interdependencies,
leaving them to be inferred by the reader. These were also static representations of a planned
schedule.
        \newline
        \newline
        The chart is named after Henry Gantt (1861-1919), who designed his chart around the
years 1910-1915.
        \newline
        \newline
        One of the first major applications of Gantt charts was by the United States during World
War 1, at the instigation In 1912, Hermann Schürch [de] published what would be considered
Gantt charts while discussing a of General William Crozier.
        \newline
        \newline
        The earliest Gantt charts were drawn on paper and therefore had to be redrawn entirely in
order to adjust to schedule changes. For many years, project managers used pieces of paper or
blocks for Gantt chart bars so they could be adjusted as needed. Gantt's collaborator Walter
Polakov introduced Gantt charts to the Soviet Union in 1929 when he was working for the
Supreme Soviet of the National Economy. They were used in developing the First Five Year Plan,
supplying Russian translations to explain their use.
        \newline
        \newline
        In the 1980s, personal computers allowed widespread creation of complex and elaborate
Gantt charts. The first desktop applications were intended mainly for project managers and
project schedulers. With the advent of the Internet and increased collaboration over networks at
the end of the 1990s, Gantt charts became a common feature of web-based applications,
including collaborative groupware. By 2012, almost all Gantt charts were made by software which
can easily adjust to schedule changes.
        \newline
        \newline
        In 1999, Gantt charts were identified as "one of the most widely used management tools
for project scheduling and control".
        \newpage
        \begin{figure}
            \centering
            \includegraphics[width=\linewidth]{ganttchart.jpg}
            \caption{Gantt Chart}
            \label{fig:my_label}
```

```
\end{figure}
\end{document}
```