REPORT 2

for

Global Calculator and Converter

Prepared by

20Z201- Aadhithyashri Manikandan

20Z214- Dhanush Gowdhaman

20Z217- Harsha Vardhan V M

20Z223- Kabilan K K

20Z239- Prednya Ramesh

20Z251- Shri Shivathmika V

PSG COLLEGE OF TECHNOLOGY

TABLE OF CONTENTS

Software Requirements Specification

for

Global Calculator and Convertor

Version 2.0 approved

Prepared by

20Z201- Aadhithyashri Manikandan

20Z214- Dhanush Gowdhaman

20Z217- Harsha Vardhan V M

20Z223- Kabilan K K

20Z239- Prednya Ramesh

20Z251- Shri Shivathmika V

PSG COLLEGE OF TECHNOLOGY

26-08-2022

Table of Contents

Revision History

Name	Date	Reason For Changes	Version
Version 2 SRS	26/09/22	Addition of extra functionalities in converter	2

1. Introduction

1.1 Purpose

The Global Calculator and Converter helps to perform calculations of complex values faster, more accessible, and efficiently. The system provides various calculators and converters such as Temperature Conversion, Numerical System conversion, Matrix computation, Mathematics and Physics formulae, Scheduling algorithms in Operating Systems, Time Conversion, Encryption and Decryption, and many other miscellaneous conversions.

1.2 Document Conventions

This document follows the IEEE template for System Requirement Specification Documents:
Alignment of text is justified
Important headers are bolded
For headings

- Times New Roman, size 14

For content

- Times New Roman, size 12

1.3 Intended Audience and Reading Suggestions

The Global Calculator and Converter is best suited for a wide range of people. Engineers and students might seldomly use certain calculators and converters. The document contains information about the various subsystems of the system.

1.4 Product Scope

The purpose of the Global Calculator and Converter is to provide a consolidated space to find as many possible calculators and converters. This system aims at providing a user-friendly interface for faster and more efficient calculations. The system's benefits are providing students with the verification of answers and performing time-consuming calculations.

2. Overall Description

2.1 Product Perspective

The existing systems provide only basic calculations in an integrated space. The Global Calculator and Converter is a multi-functional system that provides an integrated environment that can be used by students and engineers to perform complex calculations and conversions. This system consists of various calculators and converters ranging from mathematics, physics, engineering, etc. The intention of developing the system is to have cumbersome free calculations and conversions.

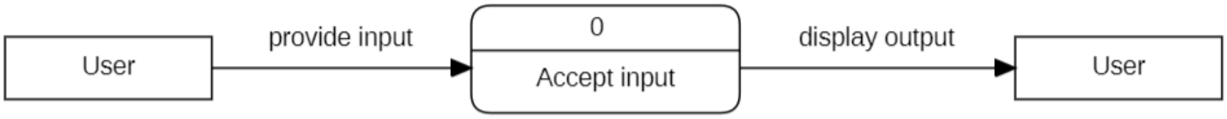
2.2 Product Functions

- The system consists of two modes: Calculator and Converter

- The Calculator mode consists of various other calculators such as
  - Matrix Computation
  - Mathematical and Physics formulae
  - Operating System - Scheduling Algorithm Computation
  - Encryption and Decryption

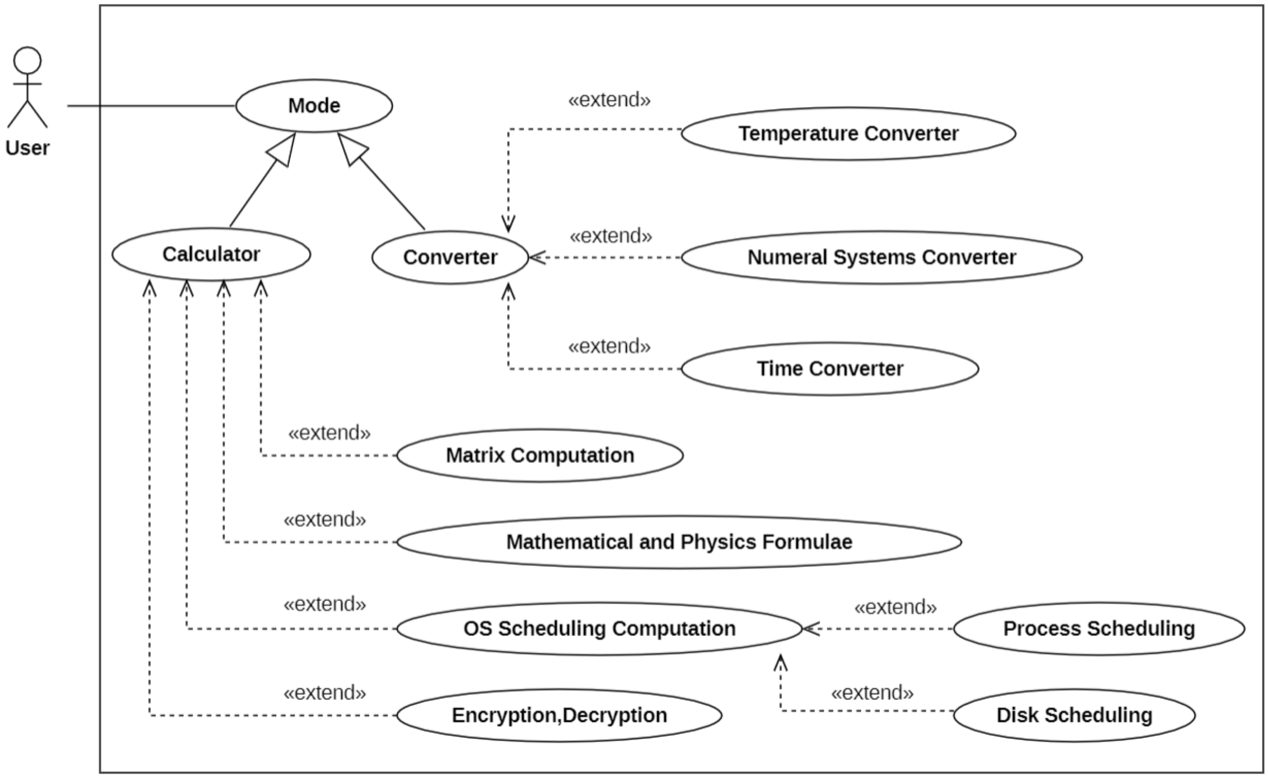
- The Converter mode consists of various other converters such as
  - Temperature
  - Numeral Systems
  - time

Data flow diagram(level 0):



2.3 User Classes and Characteristics

User can be a student, an engineer, an instructor, a graduate, etc. If the user is a student, the basic calculators, and converters such as mathematics and physics formulae will be used more frequently than the Operating system scheduling computation. On the other hand, an engineer or a graduate may use Operating system scheduling computation, Encryption and Decryption, and other complex computations more frequently than basic computations. The frequency of the use cases of the system differs based on the user classes.



2.4 Operating Environment

The operating environment would be an environment that supports python files and utilizes the suitable python compiler.

2.5 Design and Implementation Constraints

The design of the system should be in such a way that it results in no malfunctions. The system aims to provide the right output assuming that the user knows the format of the input to be given.

2.6 User Documentation

The input format for each calculator and converter will be mentioned in the documentation that can be downloaded from a web application.

2.7 Assumptions and Dependencies

It is assumed that the software works only with a proper network connection.

3. External Interface Requirements

3.1 User Interfaces

The user should be able to see the dashboard which contains different categories of calculators and converters.

3.2 Hardware Interfaces

A windows, Linux, or macOS-based computer with the capability to run a web browser. A Smartphone that can access the internet and utilize a web browser. The client will communicate via the internet using a secure HTTPS protocol, access tokens, and valid SSL certificates.

3.3 Software Interfaces

3.3.1 Python
- **3.3.1.1** HTML
- **3.3.1.2** CSS
- **3.3.1.3** JavaScript

3.3.2 Flask

3.4 Communications Interfaces

This package is focused on giving the right output when the input parameters are of the right format. The only communication that is occurring is the calling of a particular function in a python program or a REST API. These communications are handled by the python compiler or the REST Client respectively.

4. System Features

4.1 Calculator

4.1.1 Matrix Computation

Perform basic matrix operations

4.1.1.1 Addition

4.1.1.2 Subtraction

4.1.1.3 Multiplication

4.1.1.4 Rank and Nullity

4.1.2 Mathematics and Physics Formulae

4.1.2.1 Distance(Physics and Maths)

4.1.2.2 Speed (Physics)

4.1.2.3 Temperature

4.1.2.4 Pressure

4.1.2.5 Angles

4.1.2.6 BMI

4.1.3 Operating System- Scheduling algorithm computation

4.1.3.1 First-Come, First-Served (FCFS) Scheduling

4.1.3.2 Shortest-Job-First (SJF) Scheduling

4.1.3.3 Priority Scheduling

4.1.3.4 Shortest Remaining Time

4.1.3.5 Round Robin(RR) Scheduling

4.1.4 Disk Scheduling algorithms

4.1.4.1 FCFS scheduling algorithm

4.1.4.2 SSTF (shortest seek time first) algorithm

4.1.4.3 SCAN scheduling

4.1.4.4 C-SCAN scheduling

4.1.4.5 LOOK Scheduling

4.1.4.6 C-LOOK scheduling

4.1.5 Encryption and Decryption

This module takes a string input and encrypts or decrypts it

4.2 Convertor

4.2.1 Temperature

This module includes Temperature interconversions between Celsius, Fahrenheit, and Kelvin

4.2.2 Numerical Systems

This module includes interconversion in computer science number systems and units of measurement

4.2.3 Time

This module converts 12- hour time format to 24-hour time format and vice versa

4.2.4 Distance interconversion

This module inter-converts between the different units of measurement for distance

4.2.4 Speed interconversion

This module inter-converts between the different units of measurement for speed

5. Other Nonfunctional Requirements

5.1 Performance Requirements

This is a python package so it performs best on the latest stable build of python. It can be used in other applications like stand-alone applications or REST APIs whose performance depends not on this package.

5.2 Safety Requirements

This package is a simple calculator-converter package that does not need any authentication as there are no data and/or security threats. This is intended to be used as a part of larger applications and projects that are python based.

5.3 Security Requirements

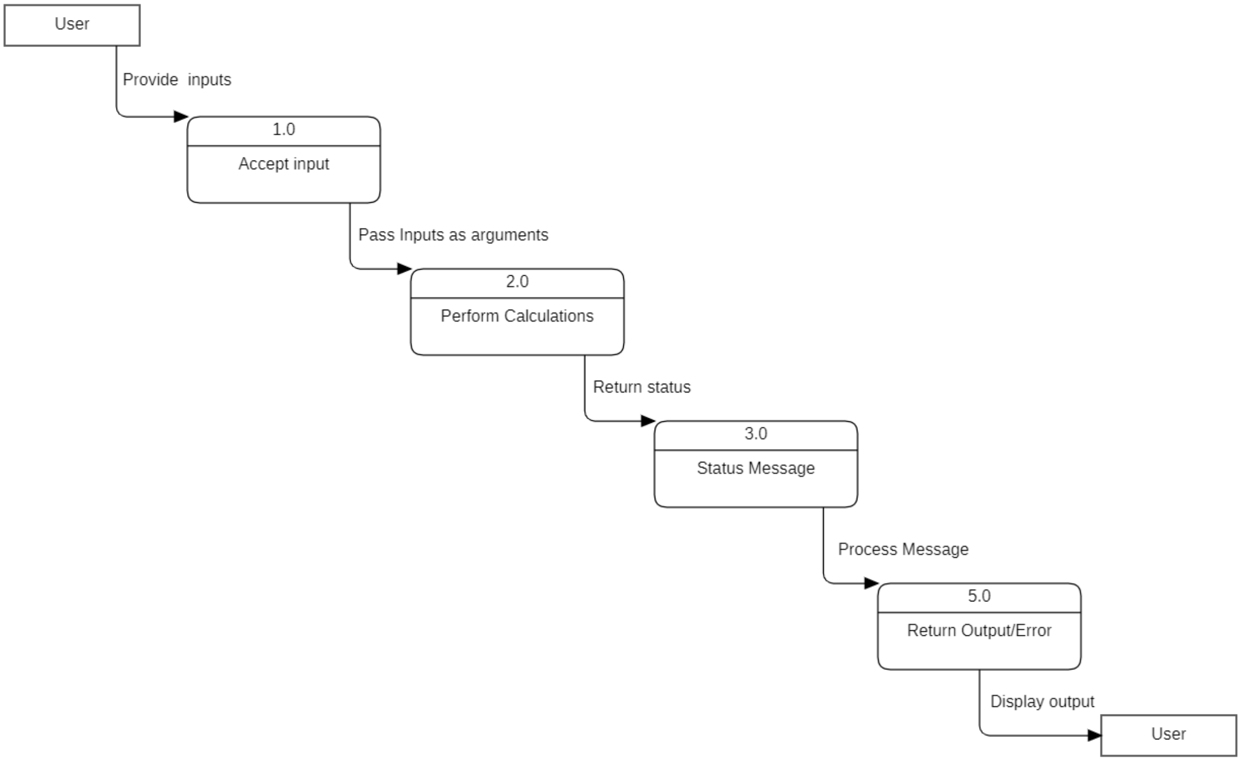
There are no special security requirements for this package as this does not access the internet for any computation other than forex which utilizes simple web scraping from a public domain like yahoo or other forex websites. This package's security needs are minimal and can be used without much hassle.

5.4 Software Quality Attributes

Software quality attributes include correctness, reliability, adequacy, learnability, robustness, maintainability, readability, extensibility, testability, efficiency, and portability.

Appendix A: Analysis Models

Data Flow Diagram:(Level 1)



Software Architecture Document

for

Global Calculator and Convertor

Version 1.0 approved

Prepared by

20Z201- Aadhithyashri Manikandan

20Z214- Dhanush Gowdhaman

20Z217- Harsha Vardhan V M

20Z223- Kabilan K K

20Z239- Prednya Ramesh

20Z251- Shri Shivathmika V

PSG COLLEGE OF TECHNOLOGY

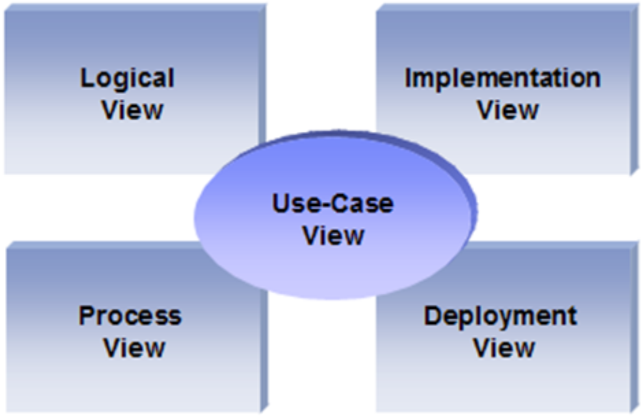
26-09-2022

Table of Contents

1.1 Purpose

The Software Architecture Document (SAD) provides a comprehensive architectural overview of the Global Calculator python package. It presents a number of different architectural views to depict different aspects of the system. It is intended to capture and convey the significant architectural decisions which have been made on the system.

In order to depict the software as accurately as possible, the structure of this document is based on the "4+1" model view of architecture.



The "4+1" View Model allows various stakeholders to find what they need in the software architecture.

1.2 Scope

The scope of this SAD is to depict the architecture of the Global Calculator Python package.

1.3 Definitions, Acronyms and Abbreviations

UML: Unified Modeling Language

SAD: Software Architecture Document

2. Architectural Representation

This document gives details of the architecture using the views defined in the “4+1” model, but using the RUP naming convention. The views used to document the Global Calculator are:

Logical view

Audience: Designers.

Area: Functional Requirements: Describes the design's object model. Also describes the most important use-case realizations.

Related Artifacts: Design model

Process view

Audience: Integrators.

Area: Non-functional requirements: Describes the design's concurrency and synchronization aspects.

Related Artifacts: (no specific artifact).

Implementation view

Audience: Programmers.

Area: Software components: Describes the layers and subsystems of the application.

Related Artifacts: Implementation model, components

Deployment view

Audience: Deployment managers.

Area: Topology: Describes the mapping of the software onto the hardware and shows the system's distributed aspects.

Related Artifacts: Deployment model.

Use Case view

Audience: All the stakeholders of the system, including the end-users.

Area: Describes the set of scenarios and/or use cases that represent some significant, central functionality of the system.

Related Artifacts : Use-Case Model, Use-Case documents

Data view (optional)

Audience: Data specialists, Database administrators

Area: Persistence: Describes the architecturally significant persistent elements in the data model

Related Artifacts: Data model.

3. Architectural Goals and Constraints

This section describes the software requirements and objectives that have some significant impact on the architecture

3.1 Technical Platform

The Global Calculator Python Package will be available to all to be downloaded on demand and used license free for any application, be it web or stand alone.

3.2 Transaction

There are no internal transactions occurring between the modules of the package.

3.3 Security

The internal security measures of the Global calculator package are only as strong as the service that implements the same. The package is a collection of useful functionalities in one common place for use by everyone. As this can be part of a stand alone application or a web application, the security measures in place are only the ones present in the implementing implementation.

3.4 Persistence

Data Persistence is property of data to exist over time. The scope of persistence is only during the execution of the function. This system has no database to store any form of data.

3.5 Availability

The Global Calculator Package will be available for download for anyone who intends to implement it in their application.

3.6 Performance

This Package is implemented using simple python functionalities that make it fast and efficient.

4. Use-Case View

This section lists use cases or scenarios from the use-case model if they represent some significant, central functionality of the package. The only use-case with a significant impact on the online catering architecture is the one related to online orders. It includes a search feature as well as a call to external services (delivery and payment).

4.1 Stand alone applications

This package can be used in stand alone applications for its individual functionalities or as a part of a stand alone application that implements most of the functionalities.It is up to the developer to decide how to use this package in their application.

4.2 As a backend to a web application

This package can be used in any python related applications as a backend for a web server. This can be easily implemented in a flask application to create RESTful APIs that connect a web application's front end and the respective back end.

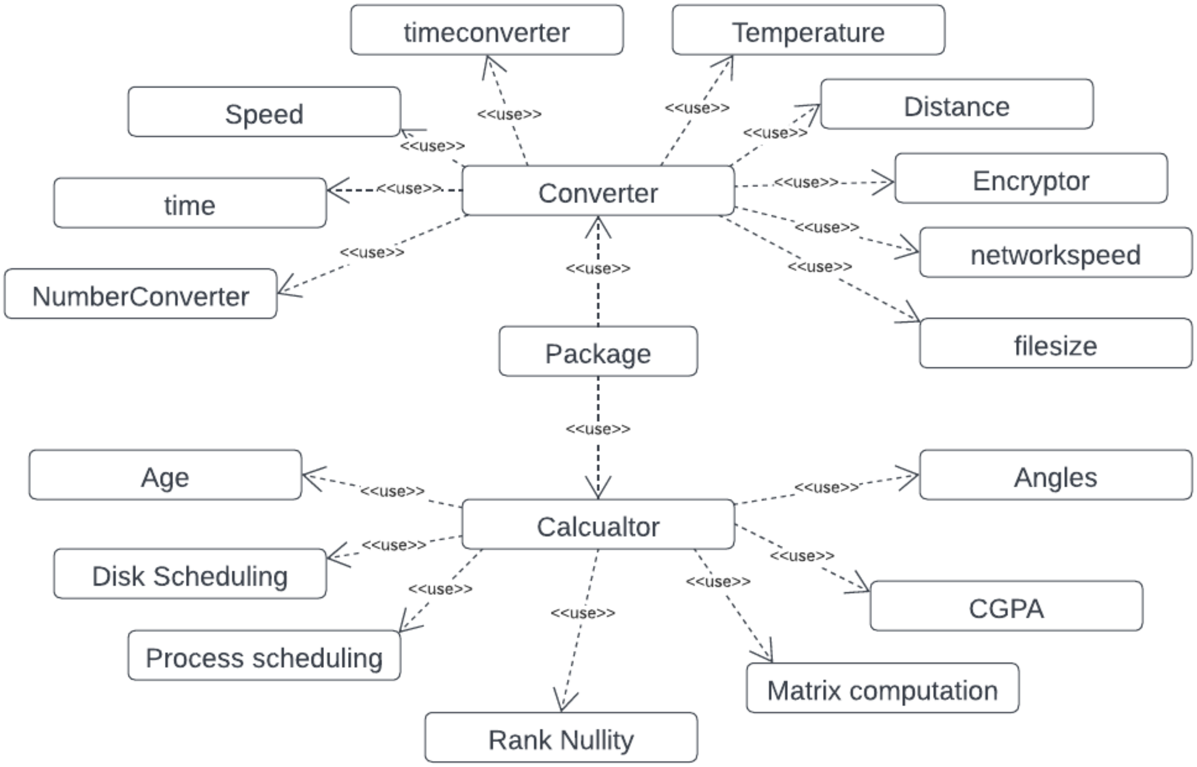
5. Logical View

5.1 Overview

The Global calculator package is divided into two main components, calculator and converter. This makes it easier to understand and implement in any application that is necessary in, be it a web application or a standalone application.

5.2 Architectural Layer Dependencies

The Global Calculator package is a simple python package which is a collection of files that can be easily used in any python application. The package.py file is the entry point into the package from which the user can use any of the available calculators or converters as they desire. The package.py file is the integration file for the package. Any new module created needs to be appended here and the file needs to be included in the package folder and the new module is now available for use.



6. Process View

There is only a simple process in place. It is a simple python function call to access any calculator or converter so any process is handled implicitly by the python interpreter or explicitly by the developer implementing the same

7. Deployment View

The Global Calculator package will be available for download for anyone interested in using it. It will initially be a simple downloadable, later converted into a pip installable package.

8. Implementation View

8.1 Overview

The Implementation view depicts the physical composition of the implementation in terms of Implementation Subsystems, and Implementation Elements (directories and files, including source code, data, and executable files).

8.2 Components

There are two main components in the package, Converter and Calculator.

8.2.1 Converter

The converter module is responsible for all computations that involve interconversion from one form to another. A good example would be the numberconverter module that can convert any binary, octal, decimal or hexadecimal number from one form to another. Network speed module is also a good example as it can take input of network speed in one unit and convert it to another.

8.2.2 Calculator

The calculator module is used to compute a certain value based on the inputs provided. An example would the angles module. It has two methods, one that calculates radians from degrees and vice versa. Age module is also another calculator module. Entering the date of birth to the age method will compute the age as of the present time.

9. Data View

The Package has no database requirements so the data view is a representation of the persistence of data, which is only during the execution of the particular functionality.

10. Size and Performance

Volume:

- Estimated degree of utilization: 100 to 10000 accesses to any method in the module.

Performance:

- Size of package: Under 90KB
- Speed of computation : Under 1 second for any module

11. Quality

The Global Calculator package has the following quality goals :

- **Scalability**: The Ability of the system to handle the load when the number of accesses increases. Since this is a module of simple nature, it is only as scalable as the application implementing the same.
- **Reliability**: The property that ensures the correctness of the result provided. The package is highly efficient and reliable.
- **Portability**: The package can be implemented in any Python-based application so its portability is limited to all python based applications.

UML Diagrams

for

Global Calculator and Converter

Version 1.0 approved

Prepared by

20Z201- Aadhithyashri Manikandan

20Z214- Dhanush Gowdhaman

20Z217- Harsha Vardhan V M

20Z223- Kabilan K K

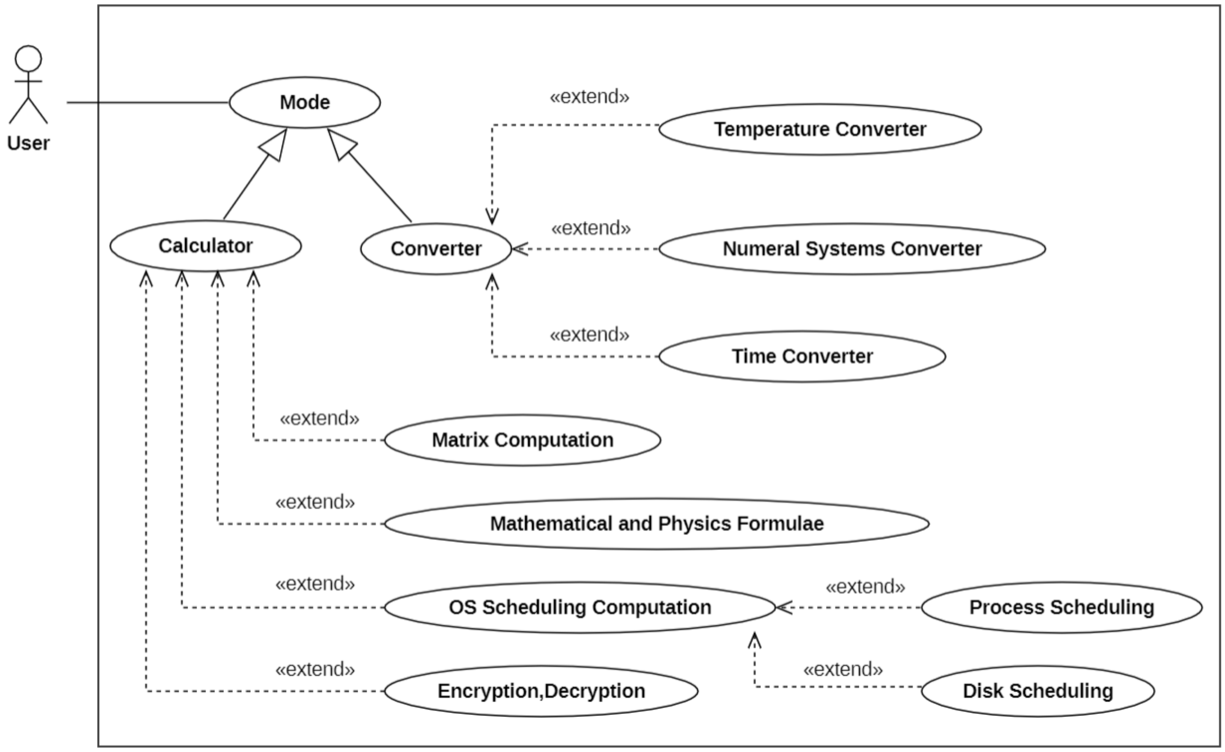
20Z239- Prednya Ramesh

20Z251- Shri Shivathmika V

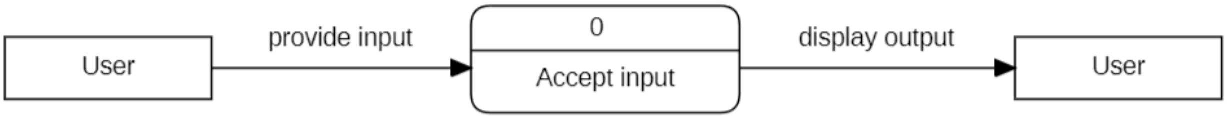
PSG COLLEGE OF TECHNOLOGY

26-09-2022

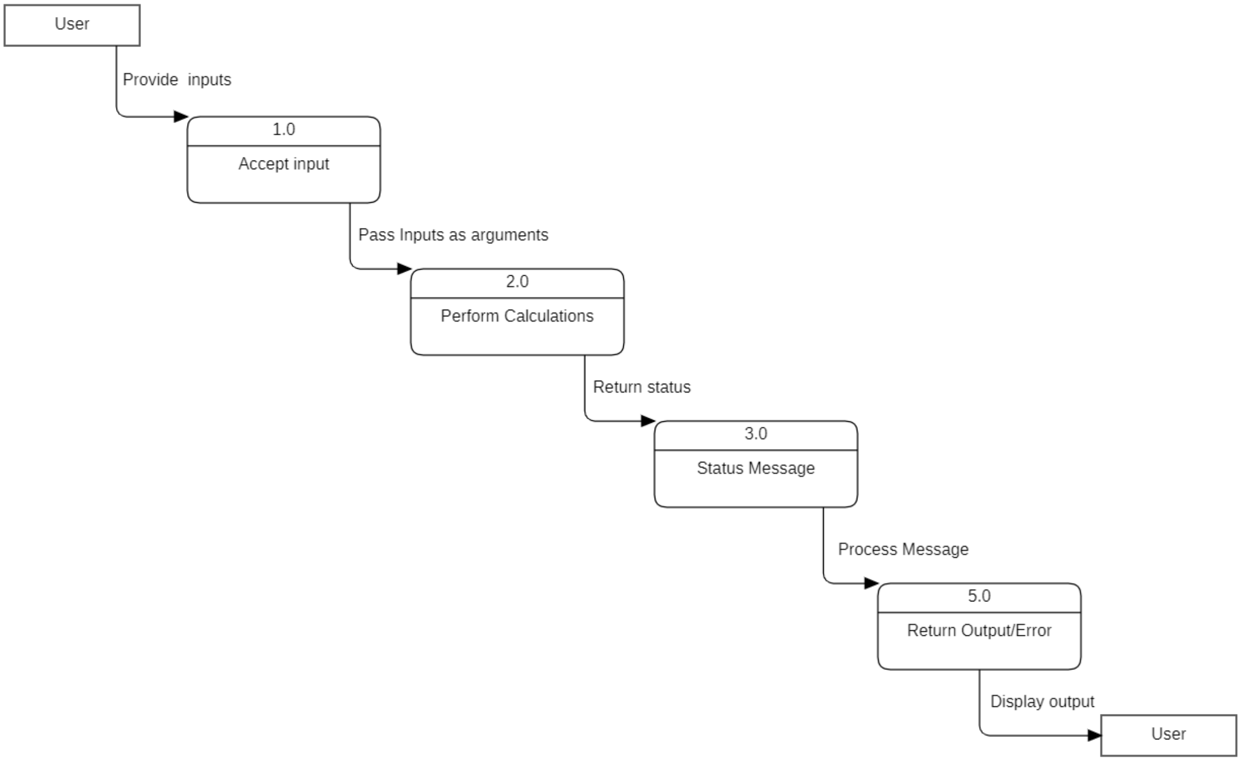
Use Case Diagram:



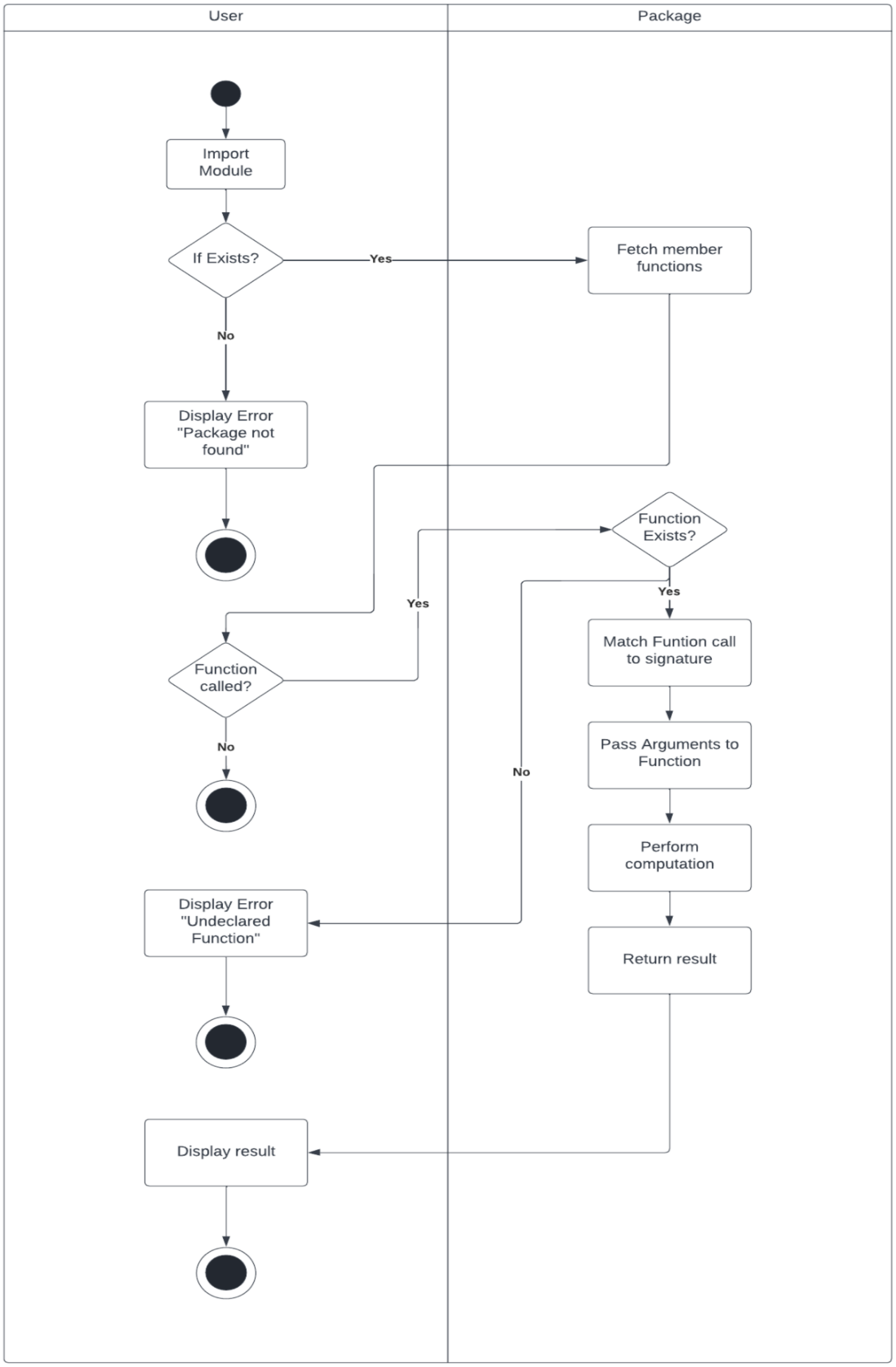
Data Flow Diagram (Level 0)



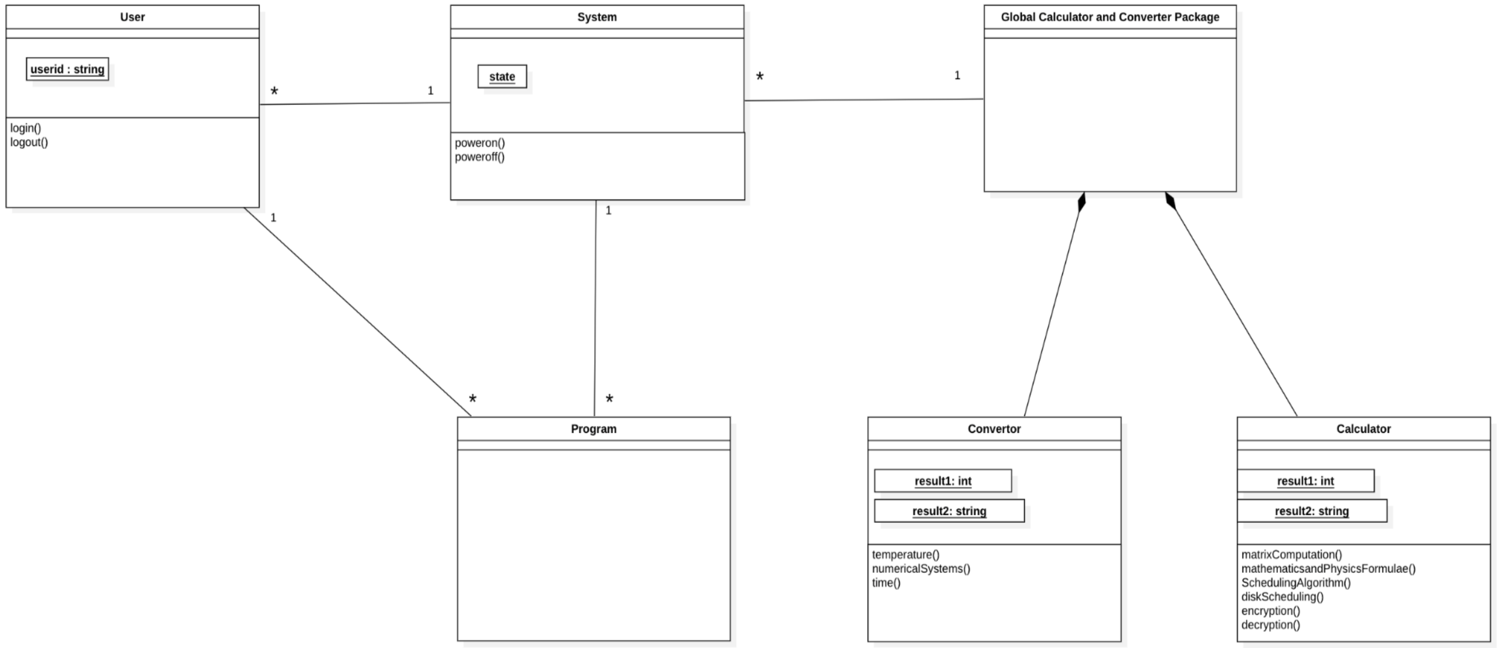
Data Flow Diagram(Level 1)



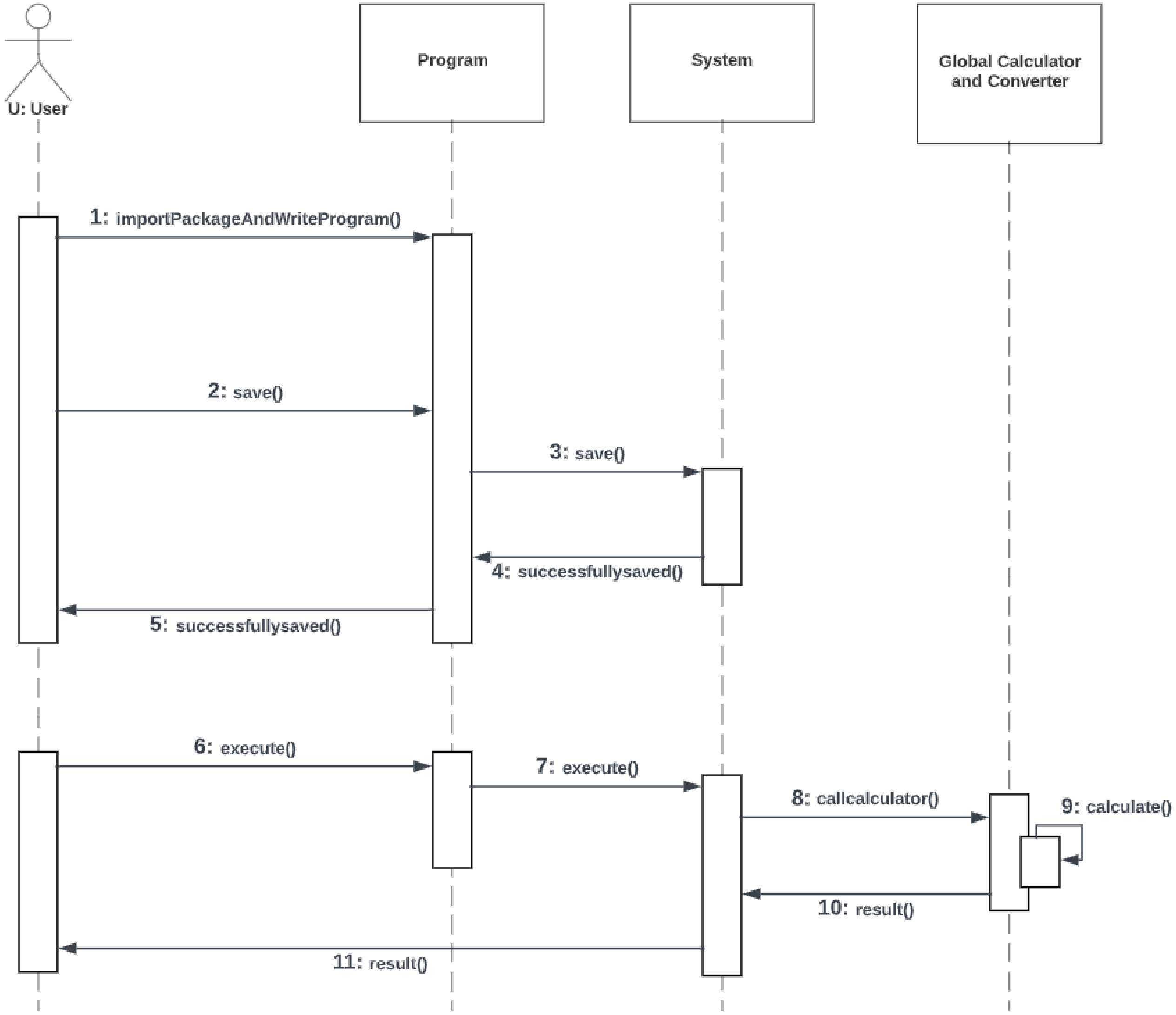
Activity Diagram:



Class diagram:



Sequence Diagram:


U: User
Program
System
Global Calculator and Converter
1: importPackageAndWriteProgram()
2: save()
3: save()
4: successfullysaved()
5: successfullysaved()
6: execute()
7: execute()
8: callcalculator()
9: calculate()
10: result()
11: result()