



**CS5002NI Software Engineering**  
**McGregor Institute**  
**20% Group Coursework**  
**AY 2023-2024**

Group Name:			
S.N.	Student Name (Section name)	College ID	University ID
1	Roshani Rauniyar (L2C18)	np01cp4s230166	22085824
2	Aditi Basnet (L2C18)	np01cp4s230042	22085490
3	Lata Kumari Thakurathi (L2C18)	np01cp4s230059	22085543
4	Aditya Shah (L2C18)	np01cp4s230031	22085814
5	Prajwal Chaulagai (L2C18)	np01cp4s230006	22085561

*I confirm that I understand my coursework needs to be submitted online via MySecondTeacher under the relevant module page before the deadline in order for my assignment to be accepted and marked. I am fully aware that late submissions will be treated as non-submission and a marks of zero will be awarded.*

## Contents

1.	Introduction .....	1
1.1	Introduction to the Business .....	1
1.2	Aims and Objectives .....	1
1.2.1	Aim .....	1
1.2.2	Objectives .....	1
1.2.3	Business Rules .....	2
2.	Project Charter .....	2
2.1	Project Charter for business .....	3
3.	Software Requirement Specifications (SRS) .....	5
3.1	Introduction .....	5
3.1.1	Purpose .....	5
3.1.2	Document Conventions .....	5
3.1.3	Intended audience .....	5
3.1.4	Project Scope .....	5
3.2	Overall Description .....	6
3.2.1	Product Perspective .....	6
3.2.2	Product Functions .....	7
3.2.3	User Classes and Characteristics .....	7
3.2.4	Operating Environment .....	7
3.2.5	Design and Implementation Constraints .....	7
3.3	Functional Requirements .....	8
3.4	Non-Functional Requirements .....	9
3.4.1	External Interface Required .....	10
3.4.2	Other Non-Functional Requirements .....	11
3.5	Goals of Implementation .....	11
4	Group Task .....	12
4.1	Environmental Model Specification .....	12
4.1.1	Data flow Diagram .....	12
4.2	Internal Model Specification .....	17
4.2.1	Entity Relationship Diagram .....	18
4.2.1.1	Entity .....	18
4.2.1.2	Attribute .....	18
4.2.1.3	Relationship .....	18

4.2.2	Data Dictionary .....	19
4.2.3	Process Specification .....	21
4.3	Design Specification.....	23
4.3.1	Structure Chart.....	23
4.5	Assignment diary .....	26
4.4.1	Assumptions.....	26
4.4.2	Omission/Inconsistency:.....	27
4.2	Group member responsibilities .....	27
4.6	Group meeting .....	29
5	Individual task .....	31
5.1	Make Payment( Prajwal Chaulagain) .....	31
5.1.1	Context Level Diagram.....	31
5.1.2	Level 1 Data Flow Diagram .....	32
5.1.3	Level 2 Data Flow Diagram .....	33
5.1.4	Structure Chart.....	34
5.1.5	Module Specification .....	35
5.2	Purchase Plant(Aditya Shah) .....	37
5.2.1	Context Level Diagram.....	37
5.2.2	Level 1 DFD .....	37
5.2.3	Level 2 DfD .....	38
5.2.4	Structure Chart.....	39
5.2.5	Module Specification for Plant Purchase:.....	39
5.3	Report Preparation(Roshani Rauniyar) .....	43
5.3.1	Context level diagram .....	43
5.3.2	Level 1 .....	43
5.3.3	Level 2 .....	44
5.3.4	Structure Chart.....	45
5.3.5	Module Specification .....	46
5.3.6	Overall Description .....	51
5.4	Join the program(Aditi Basnet).....	52
5.4.1	Context Level Diagram.....	52
5.4.2	Level 1 DFD .....	52
5.4.3	Level 2 DFD .....	53
5.4.4	Structure Chart.....	54

5.4.5 Module Specification .....	55
5.5 Take Certification Exam (Lata Kumari Thakurathi).....	56
5.5.1 Context Level Diagram.....	56
5.5.2 Level 1 DFD .....	56
5.5.3 Level 2 DFD .....	57
5.5.4 Structure Chart.....	58
6 Summary .....	60
References.....	61

## Table of figures

Figure 1 Entity relationship Diagram .....	7
Figure 2 DFD level 0 .....	13
Figure 3 DFD level 1 .....	14
Figure 4 DFD level 2(register) .....	15
Figure 5 DFD level 2 (plant purchase).....	16
Figure 6 DFD level 2(report generation).....	17
Figure 7 Entity relationship Diagram .....	19
Figure 8 Structure chart of overall DFD.....	26
Figure 9 table of individual responsibilities .....	28
Figure 10 table of group meeting .....	30
Figure 11 level 0 (make payment) .....	32
Figure 12 level 1 (make payment) .....	33
Figure 13 level 2 (make payment).....	34
Figure 14 Structure chart (make payment).....	35
Figure 15 level0(plant purchase).....	37
Figure 16 level1(plant purchase) .....	38
Figure 17 level2 (plant purchase).....	39
Figure 18 Level 0( report generation) .....	43
Figure 19 Level 1(report Generation) .....	44
Figure 20 Level 2(report Generation) .....	45
Figure 21 Structure chart(report generation) .....	46
Figure 22 level 0( Join the program) .....	52
Figure 23 level 1( Join the program) .....	53
Figure 24 level 2( Join the program) .....	54
Figure 25 Structure Chart( Join the program).....	54
Figure 26 level 0(Certification exam).....	56
Figure 27 level 1 (Certification exam).....	57
Figure 28 level 2 (Certification exam).....	57
Figure 29 structure chart (Certification exam) .....	58

## Table of tables

Table 1 table of SRS .....	4
Table 2 module specification (make payment) .....	36
Table 3 user report preparation .....	47
Table 4 expert report preparation .....	48
Table 5 financial report preparation .....	49
Table 6 Exam report preparation .....	49
Table 7 Overall report preparation .....	50
Table 8 Notification .....	51
Table 9 table of join the program .....	55
Table 13 take certificate exam .....	59

## **1. Introduction**

This module's Software Engineering assignment (CS5002NI) carries a weightage of 20% towards the final grades. It serves as an opportunity for participants to demonstrate their ability to work short-term and complete a task within a specified timeframe. The assignment involves the utilization of various tools such as Data Flow Diagrams (DFDs), Structured Charts, Entity Relationship Diagrams (ERDs), Data Dictionaries, and Process Specifications.

### **1.1 Introduction to the Business**

For seven years, the "Sparkle Botanical System" has been a leading force in the agriculture industry. Recently, they have expanded their offerings to include a top-tier education institute. Their courses in Horticulture are available at both Graduate and Undergraduate levels, and their selection of plants reflects their expertise in the field. The institute is developing a user-friendly registration system with payment options to make it easy for users to sign up for courses or purchase plants. Certification exams will be available, and users can engage in a community through posts and forums, creating a social networking site similar to LinkedIn.

### **1.2 Aims and Objectives**

#### **1.2.1 Aim**

The objective of this group coursework is to improve our practical knowledge of software project management through collaborative efforts and efficient time management.

#### **1.2.2 Objectives**

- Develop a user registration system that enables individuals to pursue certification courses or purchase plants with ease.
- Thoroughly analyze the requirements and develop a comprehensive Software Requirements Specification (SRS) that includes Data Flow Diagrams (DFDs) and Entity Relationship Diagrams (ERDs) to provide a clear picture of the system.
- Optimize the system's functionality and design its crucial components using standard procedures to ensure efficient performance.

### **1.2.3 Business Rules**

- The system allows for user registration to access the services provided.
- Users can easily enroll in courses offered by the institute through the platform.
- The system provides a convenient option for users to purchase plants, including the ability to buy in bulk if needed.
- Various payment methods are available through the system for user convenience.
- With the help of experts on the platform, users can receive location-based recommendations for plants based on soil conditions and other factors.
- The system's admin can generate detailed financial reports for both user and employees.
- Users can take certification exams for the courses they have enrolled in through the platform.
- A forum feature allows users to engage with one another, posting and voting on opinions and ideas.
- Users will receive notifications tailored to their activity on the platform.

## **2. Project Charter**

A project charter is an official document that grants the project manager permission to commence the project. It outlines their authority to collaborate with their team and utilize company resources, including personnel, technology, and finances, to bring the project to fruition. By having a project charter in place, all parties involved can gain a comprehensive understanding of the project's scope, objectives, and success metrics from the outset. The document must be endorsed by a sponsor or authoritative figure with access to funding, such as a member of the senior management team or a project leader.

(Baluch, 2023)



## 2.1 Project Charter for business

Project Charter	
<b>Name</b>	Sparkle Botanical System
<b>Description</b>	A Software is being developed for a Botanical Training Institute where users can enroll in courses and buy plants.
<b>Problem Statement</b>	McGregor Institute, a training institute specializing in agriculture courses, plans to expand into horticulture certification courses and plant sales with "Sparkle Botanical system". However, they've experienced a 20% loss increase, equating to nearly \$150,000. This hinders their ability to pay minimum wages and salaries. McGregor also lacks knowledge and planning for their desired business expansion.
<b>Business Case</b>	Sparkle Botanical System has completed extensive preparation and research to venture into the horticulture industry. The company will be providing two services - short term courses and a user forum for purchasing plants and receiving expert recommendations. To cater to a larger audience, all services will be available online, and the software is being developed accordingly. The primary objective of this business expansion is to generate a significant revenue stream to offset the losses incurred from previous investments.
<b>Goal Statement</b>	This system seeks to increase its revenue by providing additional services to current forum users. These services include taking certification courses that are completed within a short period, purchasing plants with the help of expert recommendations, and engaging with other users by commenting on or up voting their opinions.
<b>Timeline</b>	The schedule for the main phases of this project is specified below:  Planning → 2023/12/01 – 2023/12/31 Execution → 2024/01/01 – 2024/03/31

	Testing/Deployment → 2024/04/01 – 2024/06/01
<b>Scope</b>	<p>For the purpose of creating software we have defined functions fall inside and outside the scope:</p> <p><b>Inside Scope:</b></p> <ul style="list-style-type: none"> <li>- Registration for new users</li> <li>- Purchase plants</li> <li>- Various payment options</li> <li>- Short-term certification courses</li> <li>- User forum for sharing their views</li> </ul> <p><b>Outside Scope:</b></p> <ul style="list-style-type: none"> <li>- Plant delivery and courier services</li> <li>- Discounts on courses for deserving candidates</li> <li>- Various offers on plant purchase and recommendation schema for regular users.</li> </ul>
<b>Team member</b>	<ul style="list-style-type: none"> <li>- Roshani Rauniyar → Project Manager</li> <li>- Aditi Basnet → Designer</li> <li>- Lata Kumari Thakurathi → Business Analyst</li> <li>- Prajwal Chaulagain → Developer</li> <li>- Aditya Shah → QA</li> </ul>

Table 1 table of SRS

### **3. Software Requirement Specifications (SRS)**

An SRS is a crucial document that outlines the functionality and expected performance of a software product. It serves as a blueprint or roadmap for the development team, detailing the project scope and individual requirements. A well-crafted SRS covers everything from hardware interactions to software integrations, and takes into account the needs of real-life users and human interaction. Once completed, the SRS is submitted for approval, ensuring that all stakeholders are on the same page regarding the purpose and goals of the project. (Lane, 2023)

#### **3.1 Introduction**

It is the system for McGregor Institute where the users can take short-term certification courses in Horticulture. Users can purchase plants using expert recommendations as to which location supports what variety of plants based on soil quality. They can join forums where they can share their opinions and comment on others' opinions.

##### **3.1.1 Purpose**

The purpose of this project is to create an online system for registering users for certification courses and buying plants.

##### **3.1.2 Document Conventions**

The following conventions are used in this document:

- Database (DB)
- Entity Relationship Diagram (ERD)

##### **3.1.3 Intended audience**

This project is a prototype for the Botanical Training Institute's software. This project is used for user management for certification courses, plant purchase purpose, and exams and forum management.

##### **3.1.4 Project Scope**

The goal of this project is to streamline the process of gathering all required user information, registering the user, giving them the option of taking available courses of their choice, buying plants, engaging in forums, and providing a payment option. The system's data is stored in a database, which serves as the center for all users, admin,

and experts. Because of this, processes can be greatly accelerated and simplified, which makes the work of those involved easier. While supporting the current process, it is streamlined and enables for quicker and simpler decision-making.

## **3.2 Overall Description**

### **3.2.1 Product Perspective**

The following information can be stored in the system:

- **User Description**

It includes Username, Phone Number, User ID, Gender, and Location. The user is registered and can log in using their credentials and use the system and its features.

- **Course Details**

It includes the Course Name, Course Duration, Start Date, Finish Date, Certification Exams, and Marks. Users can select any short-term course they want and can move on to complete their respective certifications.

- **Plant Details**

It includes plant name, plant species, quantity, and amount.

- **Payment details**

It includes Payment ID, Bill number, Date, User ID, Course name or Plant purchased, and payment method.

- **Certification exam details**

It includes exam ID, exam name and Payment ID. This information showed details of certificate exam. User can join paid /unpaid course and sign up for certification exam.

- **Admin description**

It includes Admin ID, Designation and Admin name. The admin makes decision, get report details, and settle users results report and experts.

- **Expert description**

It includes Expert ID, Expert name, Expert phone number and Admin ID. The expert is registered to admin by using this information in the company.

### 3.2.2 Product Functions

The following ERD shows the product functions:

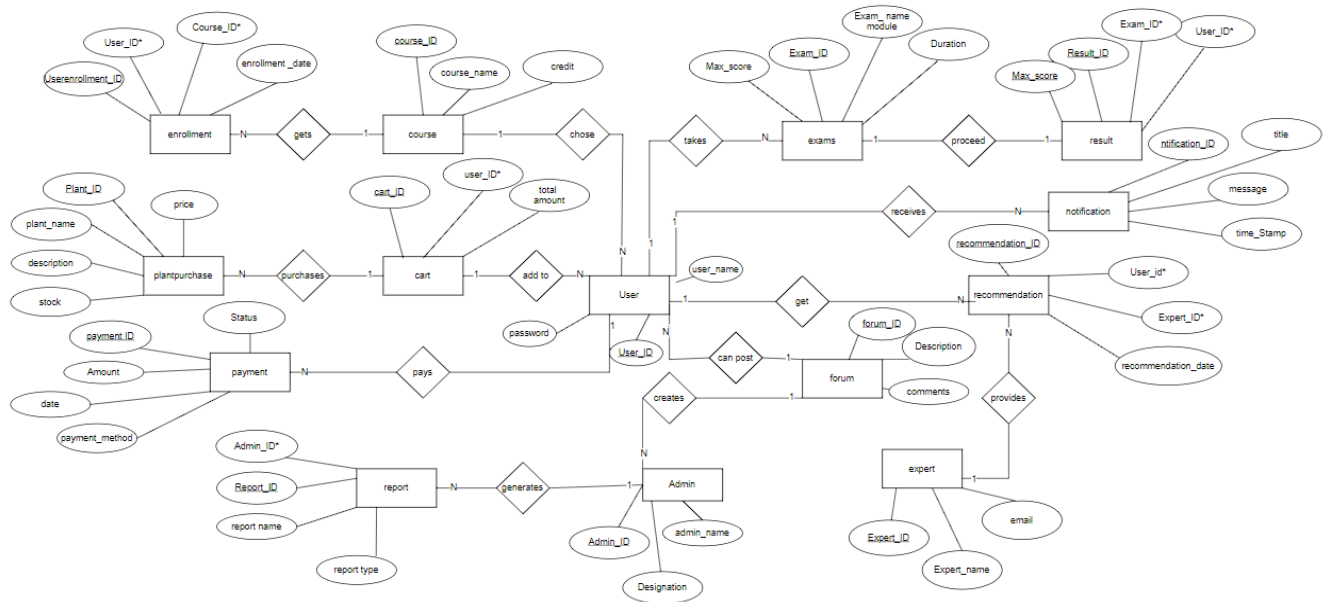


Figure 1 Entity relationship Diagram

### 3.2.3 User Classes and Characteristics

Users of the system include the Experts, system developers, and the end users. All users should be familiar with the online technologies. Users may or may not be knowledgeable about the software development process, but they should be fully aware of the goals and processes in which they are active.

### 3.2.4 Operating Environment

The operating environment for the Sparkle Botanical System is listed below:

- Platform: JAVA
- Operating System: Windows
- Database: SQL

### 3.2.5 Design and Implementation Constraints

- All the users, admin, and experts must have their information stored in a database.

- The Sparkle Botanical System is available every day of the week, readily to be used by users and experts/tutors.
- The system must have a friendly user interface.
- The system's response time should be less than two minutes.
- The software should be accessible from any device that has an internet connection.

### **3.3 Functional Requirements**

Functional Requirements are the core features that users require a system to possess as essential capabilities. These capabilities must be integrated into the system as part of the agreement, and are presented as input for the system to process, with expected output. These requirements are expressed by users and are directly observable in the final product. (Geeksforgeeks, 2023)

A brief description of the functional requirements in our system is given below:

#### **1. Register in the system**

The system can register user in the system by providing their details. The data relating to the user information is stored in a database.

#### **2. Join the program**

The user registered in the system can get enrolled in the interested course of their choice IE. Graduate or Undergraduate course. The system also allows users to join a short-term horticulture course.

#### **3. Purchase plants**

The system allows users to view available plants as per the recommendation of the experts in the forum and purchase if interested. The system allows users to keep plants in a cart and buy as per the numbers of plants checked in for.

#### **4. Payment**

The system allows users to use payment process for the plant they purchased and also the course they are enrolled in. Payment data is also stored in a database.

#### **5. Ask for recommendation**

The system provides the privilege to users to take expert recommendations as to which suitable plants or crops can be planted in particular location. The users can pinpoint their location on the map and get recommendations based on soil condition images (if possible).

## **6. Report preparation**

Admin has the facility to prepare a detailed financial report, expert(employees) report, and report related to every user including their result.

## **7. Take certification exams**

The system has section for users to sign up for tests exams at their convenience and also check results. The system lets Users to take certification exams after fulfilling certain prerequisites.

## **8. Forum**

Forum interaction allows users to engage in conversations about plants, sharing their views through posts. In the system, users can comment on and up-vote each other's' posts.

## **9. Get notifications**

The system allows all users and employees (experts) to receive relevant notifications based on their activity in the system.

## **3.4 Non-Functional Requirements**

Non-functional requirements are the quality constraints that the system should fulfill. The system's adherence to specific quality constraints is detailed in the project contract. However, the priority or importance of these factors may vary depending on the project at hand. These requirements, commonly referred to as non-behavioral requirements, encompass a wide range of aspects including

security, maintainability, reliability, scalability, performance, and flexibility. (Geeksforgeeks, 2023)

### **3.4.1 External Interface Required**

#### **1. User Interface**

A company needs to have a user-friendly interface for computer/server interaction. Creating such an interface can be done with the help of software like Figma, Dreamweaver, WordPress, and others. The website's interface should be modern and simple, making it easy for users to navigate through and use the website.

#### **2. Hardware Interface**

The hardware of a computer includes its physical components that can be touched and manipulated. The interface, on the other hand, enables users to communicate with us. To run a high-end server, it is important to have a powerful server computer. The minimum hardware specifications for such a computer are as follows: The processor should be at least Intel Celeron or higher. The processor should not exceed 5Ghz, as this can cause overheating of the CPU. The RAM should be greater than 4GB. The hard disk should have a capacity of at least 60GB.

#### **3. Software Interface**

Our system has the capacity to communicate with its users for account registration through HTTPS (Hyper Text Transfer Protocol Secured). This system allows users to take certification courses and exams, and buy plants if interested. For the database handling, we use SQL.

#### **4. Communication Interface**

Communication between the client and the server computer is called HTTPS (Hyper Text Transfer Protocol Secured). There are many Internet protocol version like IPV6 and IPV4. Being the modern and yet premium company we are, we use IPV6 as the logical address.



### **3.4.2 Other Non-Functional Requirements**

#### **1. Performance Requirements**

A wide range of certification courses are to be handled by the system. It shouldn't take long to respond. The maintenance of the system and required updates should be monitored by the respective people. The end users should be provided with up-to-date information on the desired courses.

#### **2. Safety Requirements**

There should be timely data backup. If any damage takes place to the existing data, the backup should be ready and the post-processing of the system should not be hampered in any way.

#### **3. Security Requirements**

User information and the database stored in the system should be stored securely. If necessary, we should invest in a private or a hybrid cloud infrastructure to be secure about our data security and the risks surrounding it.

#### **4. Software Quality Attributes**

The online system should be available every day of the week, all day long. The system and user should provide correct information about the requirements for registering training course. The system should meet the users' needs and must be properly maintained.

### **3.5 Goals of Implementation**

The goals of implementation of this Sparkle Botanical System is to conduct courses and run their business online. The software developer should have some sets of formal logic to create a suitable system for the institute. Both the functional and non-functional needs are implemented to improve and increase system performance. Since the system is platform independent, this implementation aim also addresses the problems that were resolved throughout system development and provides adequate documentation for system extension.

## **4 Group Task**

### **4.1 Environmental Model Specification**

A document that lists the essential attributes and specifications of an environmental model is called an environmental model specification. Ecosystems, the climate, and the effects of human activity on the environment are examples of natural systems that are represented mathematically or computationally as environmental models.

#### **4.1.1 Data flow Diagram**

Among the first types of graphical software models were data-driven models. Data-flow diagrams (DFDs) were first introduced in the 1970s as a means of representing the processing steps in a system by structured methods like DeMarco's Structured Analysis (DeMarco, 1978). Data-flow models are useful because they help analysts and designers understand what's going on by tracking and documenting the movement of data related to a specific process through the system.

Data-flow models track and record the flow of data related to a particular process throughout the system, assisting analysts and designers in understanding what's happening. Since data-flow diagrams are frequently straightforward and easy to understand, future system users can take part in model validation.

##### **4.1.1.1 Context Level Diagram (level 0)**

The highest level of DFD is the Context Level Diagram. One chart that shows every requirement for the system as well as every data flow into and out of it exists. This step displays all external entities, and it only contains one process.

To give a general overview of the system and its surroundings at the outset of the system design process, a context-level diagram is usually employed. It is an effective tool for figuring out the primary parts and procedures that comprise the system as well as for comprehending the general layout and operation of the system.

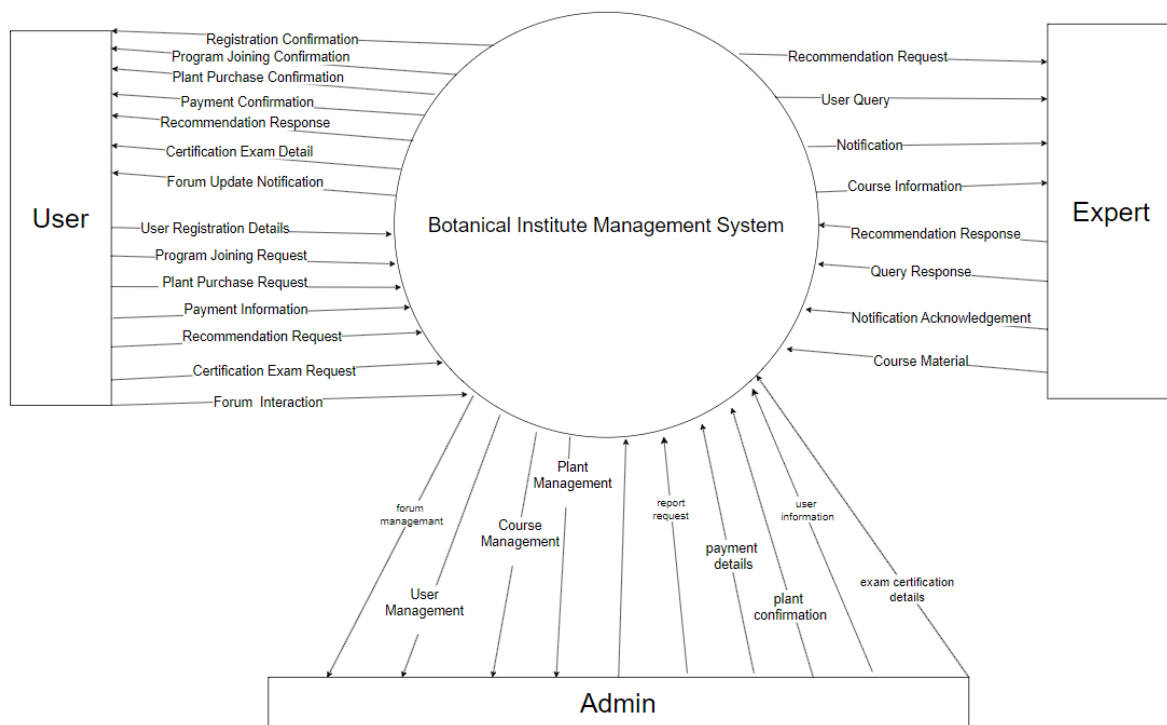


Figure 2 DFD level 0

#### 4.1.1.2 Level 1DFD

Level 0 is perceived by Level 1 DFD. It is a visual depiction of how data moves through a system. It offers a high-level overview of the system, displaying its inputs and outputs along with the main operations and data repositories that comprise it. It shows every data store that the system uses.

A Level 1 DFD uses oval shapes to represent the inputs and outputs, rectangles to represent the data stores, and squares to represent the processes. The data flow between the various components of the system is depicted with arrows. (geeksforgeeks, 2023)



Confirmation,' affirming the user's successful registration and login, ensuring a secure and streamlined onboarding experience.

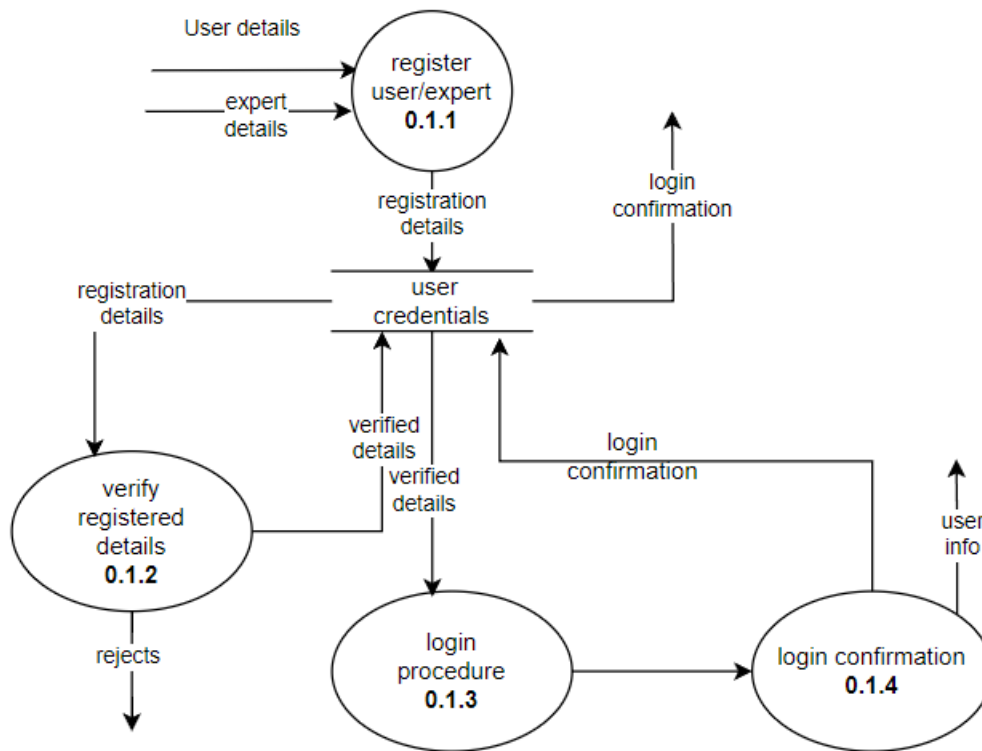


Figure 4 DFD level 2(register)

### Level 2 DFD(plant purchase)

At the second level of the 'Plant Purchase' process, three distinct sub-processes further refine the user journey. First, the 'Search Plant' sub-process enables users to explore and filter plant varieties based on preferences. Subsequently, users engage in the 'Add to Cart' step, selecting desired plants and populating a virtual shopping cart. The 'Purchase Confirmation' sub-process allows users to review their selections before confirming the purchase. Finally, the 'Payment' sub-process securely handles transaction details, completing the plant acquisition process. This detailed Level 2 DFD ensures a nuanced understanding of the sequential steps involved in the plant purchase workflow within the broader system.

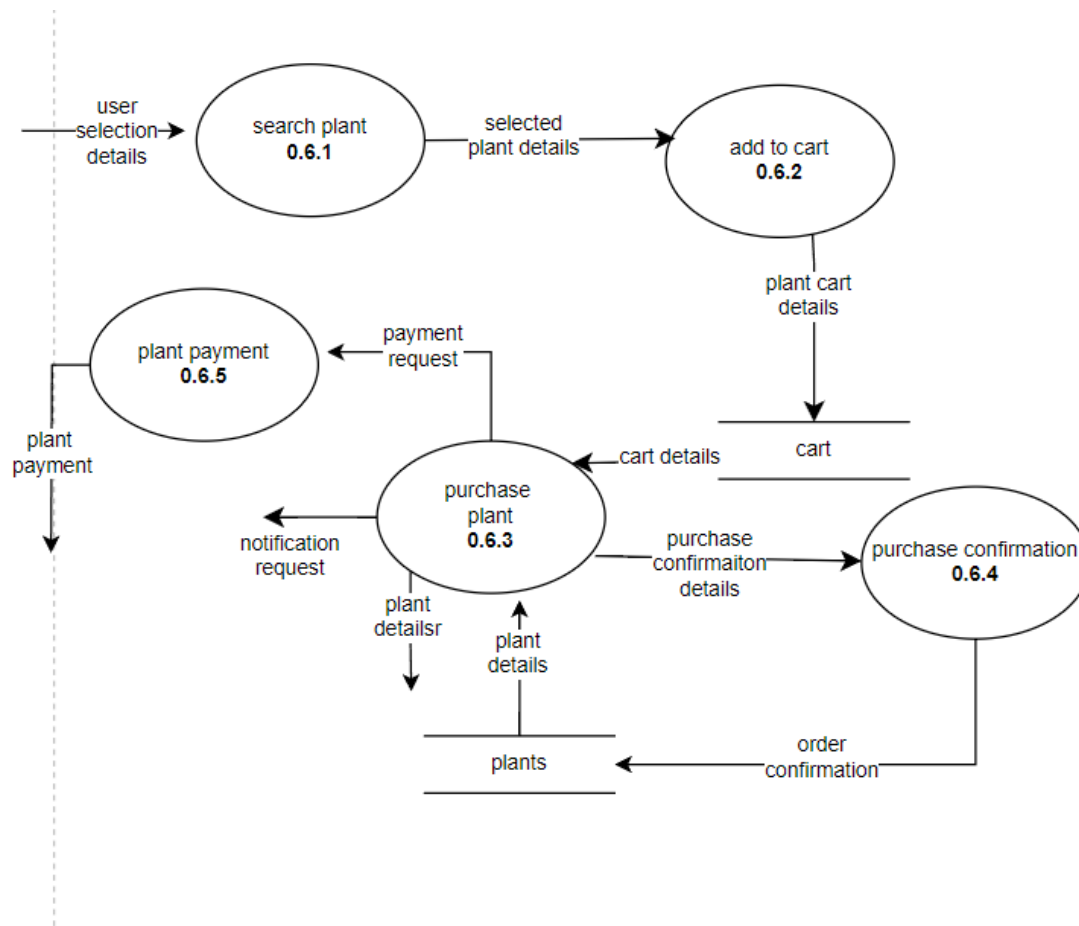


Figure 5 DFD level 2 (plant purchase)

## Level 2 DFD( report generation)

The outlined diagram delineates the report preparation process, illustrating the flow of details from various sources. Commencing with user, expert, payment, and result data, each stream feeds into its designated sub processes: 'User Report Preparation,' 'Expert Report Preparation,' 'Financial Report Preparation,' and 'Result Report Preparation.' The synthesized individual reports from these sub processes collectively progress to the 'Overall Report Generation.' This streamlined approach ensures the seamless creation of comprehensive reports, integrating insights from diverse data channels for a holistic overview.

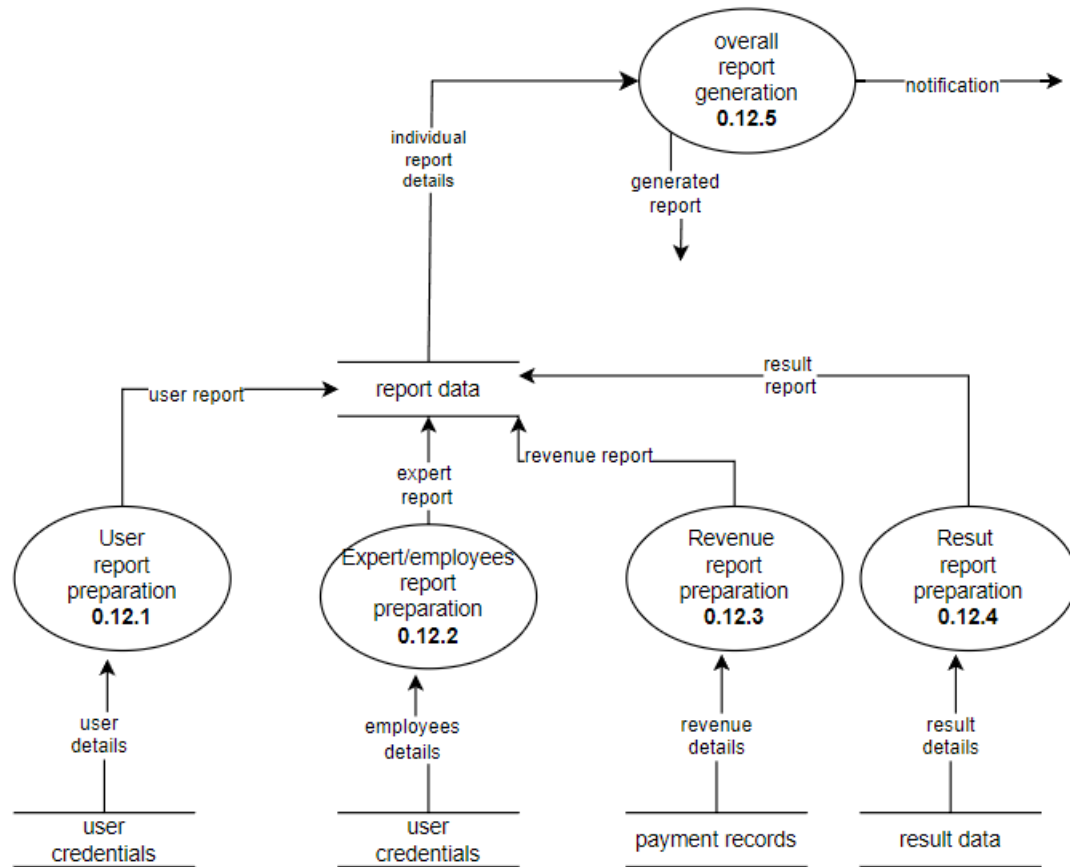


Figure 6 DFD level 2(report generation)

## 4.2 Internal Model Specification

Internal Model Specification provides a detailed representation of the system's structure and data organization. It includes components like Entity-Relationship Diagram (ERD), Data Dictionary, and Process Specifications (Pspecs). This specification delineates the relationships between entities, their attributes, and the processes that manipulate the data, fostering a comprehensive understanding of the system's internal workings. The ERD captures entity relationships, the Data Dictionary defines data elements, and Process Specifications outline the elementary processes involved, collectively forming a holistic view of the system's internal structure. (MAR31, 2023)

### **4.2.1 Entity Relationship Diagram**

An entity-relationship diagram (ER diagram) illustrates this. A collection of entities is a collection of related entities, each of which may have characteristics. Entity-relationship diagrams are frequently employed in conceptual and structured analysis modeling. The ER approach is simple to comprehend, effective at simulating real-world issues, and readily adaptable to database schemas. According to ERD, the real world is made up of a variety of business entities, their connections with one another, and the characteristics that define them. (S, 2023)

#### **4.2.1.1 Entity**

If the data is kept in a database, entities serve as representations of the data's constituent parts. This could be actual or hypothetical, living or non-living. People, places, things, events, or concepts are a few examples of entities. Entities are typically shown in an ER diagram as rectangles with the entity name at the top.

#### **4.2.1.2 Attribute**

The characteristics of an object or relationship are described by its attributes. A shipment entity, for instance, may have the following attributes: Address, User\_ID, etc.

#### **4.2.1.3 Relationship**

In the context of the McGregor Institute's botanical system, interactions between different components are crucial. For example, users establish connections with the system's entities, such as courses and plants, through actions like enrollment and plant purchase. Therefore, the relationship between these entities is established through actions like "enrollment" and "purchase." Cardinality in this scenario signifies how often a user or a course is linked to instances of other entities, demonstrating the frequency of engagement between users and courses or users and plants in the system.



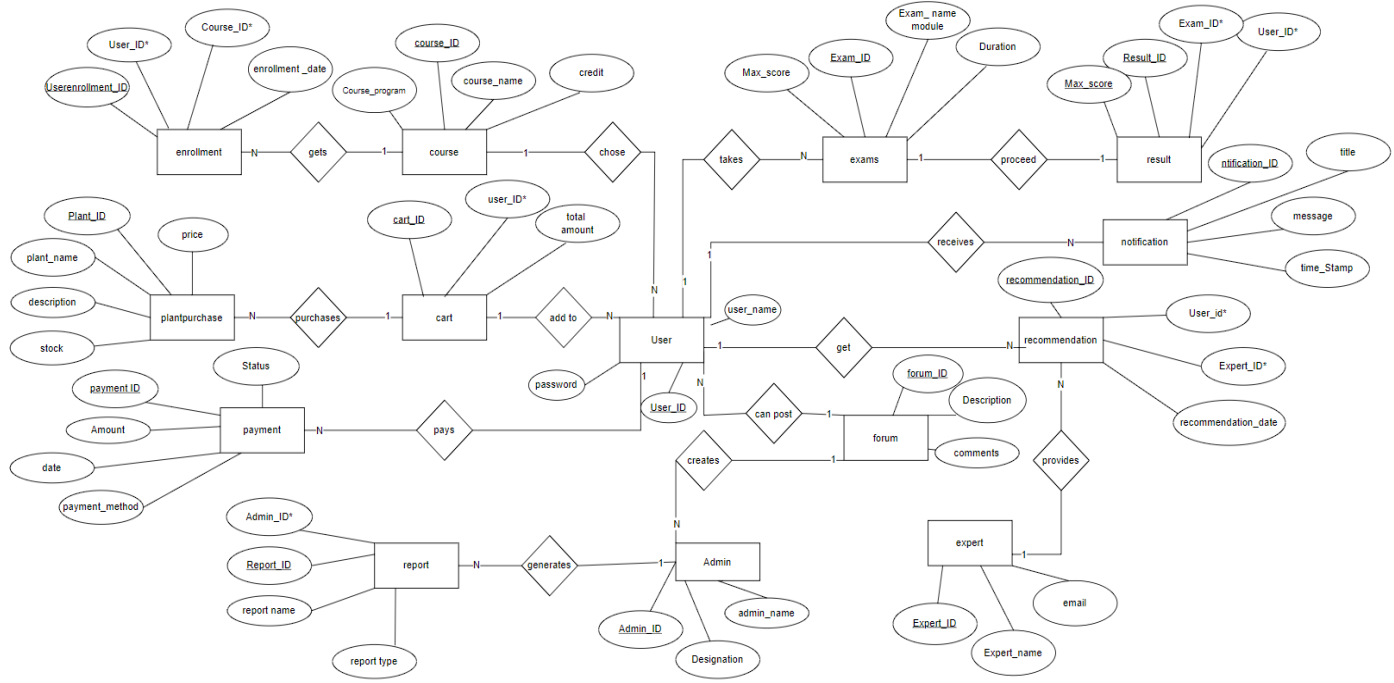


Figure 7 Entity relationship Diagram

## 4.2.2 Data Dictionary

A data dictionary is used to keep track of every business requirement for every system, as well as how those requirements are implemented into different computer systems to meet organizational needs. Records of other objects in the database, including who owns the data and how it relates to other objects, are kept in a data dictionary. (Chia, 2023)

An essential component of a relational database is a data dictionary. Ironically, most database users are unaware of it due to its significance. Fields like entity name, entity definition, attributes, names and their descriptions, data types, lengths, and constraints like primary and foreign keys are all included in the data dictionary.

The data dictionary for the data flow, data store is listed below:

Data dictionaries are useful for many purposes. To put it briefly, they:

- i. Assist in preventing disparities in data across a project.
- ii. Participate in the creation of project-wide protocols.
- iii. Ensure consistency in the way that members of a research team gather and use data.
- iv. Simplify data analysis.

v. Establish obligatory Data Standards.

The data dictionary of the Sparkle Botanical System are stated below:

- Registration :Command
- Login : Command
- Enrollment : Command
- Course : Command
- Manage Payment : Command
- Purchase Plant : Command
- Ask recommendation : Command
- Manage forum : Command
- Conduct exam : Command
- Result credentials : Command
- Notification : Command
- Enrollment : Userenrollment\_ID + User\_ID + Course\_ID+ Enrollment\_date
- Course :Course\_ID + Course\_Name + Credit
- Exams : Max\_Score + Exam\_ID + Exam\_Name Module + Duration
- Result : Result\_ID + Exam\_ID + User\_ID
- Enrollment: Command
- plant Payment: Integer
- course Payment: Integer
- plant details: plantD + userID + User info
- Queries Request: User\_ID + User\_name + Expert\_ID + queries Details
- Details for registration: Command
- Login: Command
- Login: User\_ID + Expert\_ID
- Confirmation with login details: User\_ID + user\_info
- exam Payment: Integer
- recommendation: String
- conduct exam : Command
- result credentials Confirmation: Command

- Search plants: Command
- Report Detail: String + (Revenue + userdetails+ exam records)
- Generate Report: Command
- User Info: User\_ID + User\_name + Phone\_number
- Login Credentials: User info + expert info

### **4.2.3 Process Specification**

A process specification is a vital method for documenting, analyzing, and explaining the decision-making logic and formulas utilized to generate output data from process input data. Its primary purpose is to accurately specify regulatory and engineering requirements and procedures. When it comes to producing high-quality, reliable data, having clear and comprehensive process specifications is essential. By minimizing ambiguity, a process specification provides an individual or organization with a precise description of executed tasks and accomplishments, allowing them to validate system design, including the data dictionary and data flow diagrams. (Rouse, 2012)

#### **4.2.3.1 Process Specification for elementary processes**

##### **Process A:**

Process Number: 0.1.1

Process Name: Registration.

Description: This process registers new users.

Input data flow: Receives user details.

Output data flow: Registered details.

Process logic:

- User provides credentials to the system.
- The user credentials is stored in a database of the system.
- The stored information is used for login confirmation.

##### **Process B:**

Process Number: 0.1.2

Process Name: Login procedure

Description: This process logs in the already registered users.

Input data flow: Receives expert registration details and user registration details.

Output data flow: User info and login details.

Process logic:

- User logs in using their credentials.
- User can now enroll in courses.
- Expert can also login and use the system functions and features.

### **Process C:**

Process Number: 0.6.5

Process Name: Plant payment.

Description: This process proceeds the users into the payment section.

Input data flow: Receives plant payment request.

Output data flow: Plant payment.

Process logic:

- Plant payment request is proceeded towards plant payment
- Plant payment moves towards payment process.

### **Process D:**

Process Number: 0.6.3

Process Name: Purchase plants.

Description: This process manages the feature to purchase plants.

Input data flow: Selected plant and searched plant.

Output data flow: Notification request, plant payment and order confirmation.

Process logic:

- Selected plant details is sent to the cart.

- Plant order confirmation is sent to the database. User receives confirmation notification.

### **Process E:**

Process Number 0.12.1

Process Name: User report preparation

Description: This process allows report to be prepared from the users details stored in the database.

Input data flow: User's details

Output data flow: User's report

Process logic:

- The users details are taken out from the data-store of user credentials then the details further move to report preparations section.
- The prepared individual user report further proceed to report generation.

## **4.3 Design Specification**

Design Specification outlines the structure and organization of the system through a Structure Chart. It provides a visual representation of the hierarchical relationships among processes, functions, data stores, and data flows. The upper-level Structure Chart presents an overview of the entire system, elucidating the flow of control and data. This specification is crucial for analyzing and enhancing the efficiency and effectiveness of the system's design.

### **4.3.1 Structure Chart**

A structure chart (SC) is a diagram used in software engineering that shows how the configuration system is divided into its most controllable parts. Program modules are arranged in a structure resembling a tree in a structure chart. The name of each module is displayed as a box, which facilitates understanding the relationships between the modules. In software development, structure charts are frequently used to illustrate a program's hierarchy and the connections between its various modules, subroutines, and

functions. They can also be used to demonstrate the chain of command and the connections and interactions between various teams or departments in business and organizational contexts.

Typically, structure charts are made with lines or arrows to indicate the relationships between the various system components and boxes or rectangles to represent the various system components. There is a hierarchy among the boxes, with higher-level elements at the top of the chart and elements at a lower-level underneath them. The relationships between the component are represented by the lines joining the boxes.



## 4.5 Assignment diary

The assignment diary includes all of the assumptions made during coursework completion as well as any omissions made during group discussions and the distribution of group responsibilities.

### 4.4.1 Assumptions

Our system has been meticulously crafted to meet the outlined objectives, upholding the industry's highest standards. To bolster the architecture and logic of the system, certain assumptions were made during the design process. Our team worked collaboratively to establish a set of guidelines for designing within the Sparkle Botanical System.

The assumptions for the database to store the system data are as listed below:

**Services:** All the services offered by the Sparkle Botanical System are stored in this database, including the courses and the plant purchase facility and others.

**Courses:** All the courses that are offered in the system are stored in the database.

**Experts:** The database contains the information of the experts that provide recommendations.

**User:** Personal details of the user and registered data in the system are stored in the database.

**Admin:** All the administrative information related to the system is functioned accordingly. Admin lets out the exam dates and details. Admin is responsible for making reports .

**Plants:** The details regarding the plant selection, including the plant name, variety, number in stock, and suitable environment are stored in the database.

The following requirements were analyzed by the team from the specifications based on which the above assumptions are made:

- The registration records are maintained.
- Users must choose the courses they want to enroll in.



- Courses offered and selected are based on the needs of the individual users.
- Admin maintains the payment details of the users and purchased plants procedures.
- Admin generates the report of the financial, expert and user-related.
- Users can enroll in undergraduate or graduate courses or short-term horticulture courses.
- Admin can update the record of the users and also perform data manipulation of the database that stores all the information.

#### **4.4.2 Omission/Inconsistency:**

It is important to keep in mind that during the initial stages of the application, there may be some inconsistencies that could have an impact on the business process. These inconsistencies may manifest in the following ways:

- Unrealistic deadlines may be set, which could hinder the timely completion of the project. It is crucial to establish a project timeline that is both acceptable and realistic, especially when considering the project's complexity and scope.
- Including outdated or incorrect information in the specification can have negative consequences. It is vital to confirm that all information in the specification is up-to-date and accurate.
- Requirements that are inconsistent or contradictory can cause issues, so it is essential to ensure that all requirements are consistent.
- A lack of resources, even when allocated according to the project plan, can still cause delays in completing the project.
- Improper revenue distribution may occur if the projected revenue exceeds the designated limit. This can lead to an inconsistent distribution of revenue, which is something to be mindful of.

## **4.2 Group member responsibilities**

Each member made an active coordinated effort while in charge of work. It's about planning the system. For better understanding for each team member, and the system

workflow, a lot of communication between the team members was made to evaluate their individual work.

Responsibilities of the individual members of the group are listed as follows:

Group member	Responsibilities
Roshani Rauniyar	<ul style="list-style-type: none"> <li>• Worked on individual task for Report Preparation.</li> <li>• Worked on the structure charts.</li> <li>• Contributed to the assignment diary part of the project.</li> <li>• Gave team members counsel, instructions, and direction.</li> <li>• Kept an eye on the work being done and refocus the project.</li> <li>• Contributed to the software requirement specification.</li> </ul>
Aditi Basnet	<ul style="list-style-type: none"> <li>• Worked on individual task for Join the program.</li> <li>• Worked on software requirements specifications.</li> <li>• Mocked the DFD levels of the whole system.</li> <li>• Wrote the SRS.</li> </ul>
Aditya Shah	<ul style="list-style-type: none"> <li>• Worked on individual task Purchase Plant.</li> <li>• Worked on project charter.</li> <li>• Contributed on DFD levels</li> <li>• Contributed on data dictionary.</li> </ul>
Lata Kumari Thakurathi	<ul style="list-style-type: none"> <li>• Worked on individual task for take certification exam.</li> <li>• Worked on process specification</li> <li>• Worked on designing the ERD Diagram.</li> <li>• Contributed to the introduction part of the report.</li> <li>• Worked on data dictionary.</li> </ul>
Prajwal Chaulagain	<ul style="list-style-type: none"> <li>• Worked on individual task for make payment.</li> <li>• Contributed to the ERD Diagram.</li> <li>• Contributed on process specification.</li> <li>• Contributed on writing assignment diary.</li> </ul>

Figure 9 table of individual responsibilities

## 4.6 Group meeting

We members of the group had several group meeting to discuss the various portion of the project task.

Date	Time	Location	Discussions
5/12/2023	9am to 10am	London Block	We thoroughly reviewed the problem with the group project on the first day and divided up the work among the individuals in the group.
10/12/2023	2pm to 3pm	London Block	On the second day, the team members again scheduled the work plan, and requirements were looked over. We discussed the requirements of the Sparkle Botanical System.
14/12/2023	12am to 1pm	Nepal Block	Our module teacher met with us to go over any questions we had about the prerequisites for constructing the system-wide Data Dictionary and DFD.
20/12/2023	11am to 12am	London Block	Every team member focused their entire concentration on the task at hand. We talked about each other's designs prior to improvisation.
24/12/2023	2pm to 3pm	London Block	The ERDs and DFDs were made by us.
29/12/2023	9am to 10am	Skill Block	We completed every individual task one at a time till the complete report was completed.
31/12/2023	2pm to 3pm	London Block	Our module instructor and we got together to discuss the difficulties we faced.

3/1/2024	10am to 11am	Kumari Block	We meticulously detailed the review and structured the report.
----------	--------------------	--------------	--

*Figure 10 table of group meeting*

## **5 Individual task**

The task which are done by the individual members of the group are displayed below along with environmental, internal and design specification. The tasks completed by each group member are listed below, along with the architectural, internal, and environmental specifications.

### **5.1 Make Payment( Prajwal Chaulagain)**

The institute provides the provision for the customer to pay of the plant they are interested in and also to pay for the course they want to enroll in.

#### **5.1.1 Context Level Diagram**

A context diagram also known as the level 0 diagram provides the broad overview of an information system. It is the most crucial diagram in data flow diagram and with the help of context diagram we can make level 1 and level 0 data flow diagram. It consists of only one process unlike level 1 and level 2 diagram.

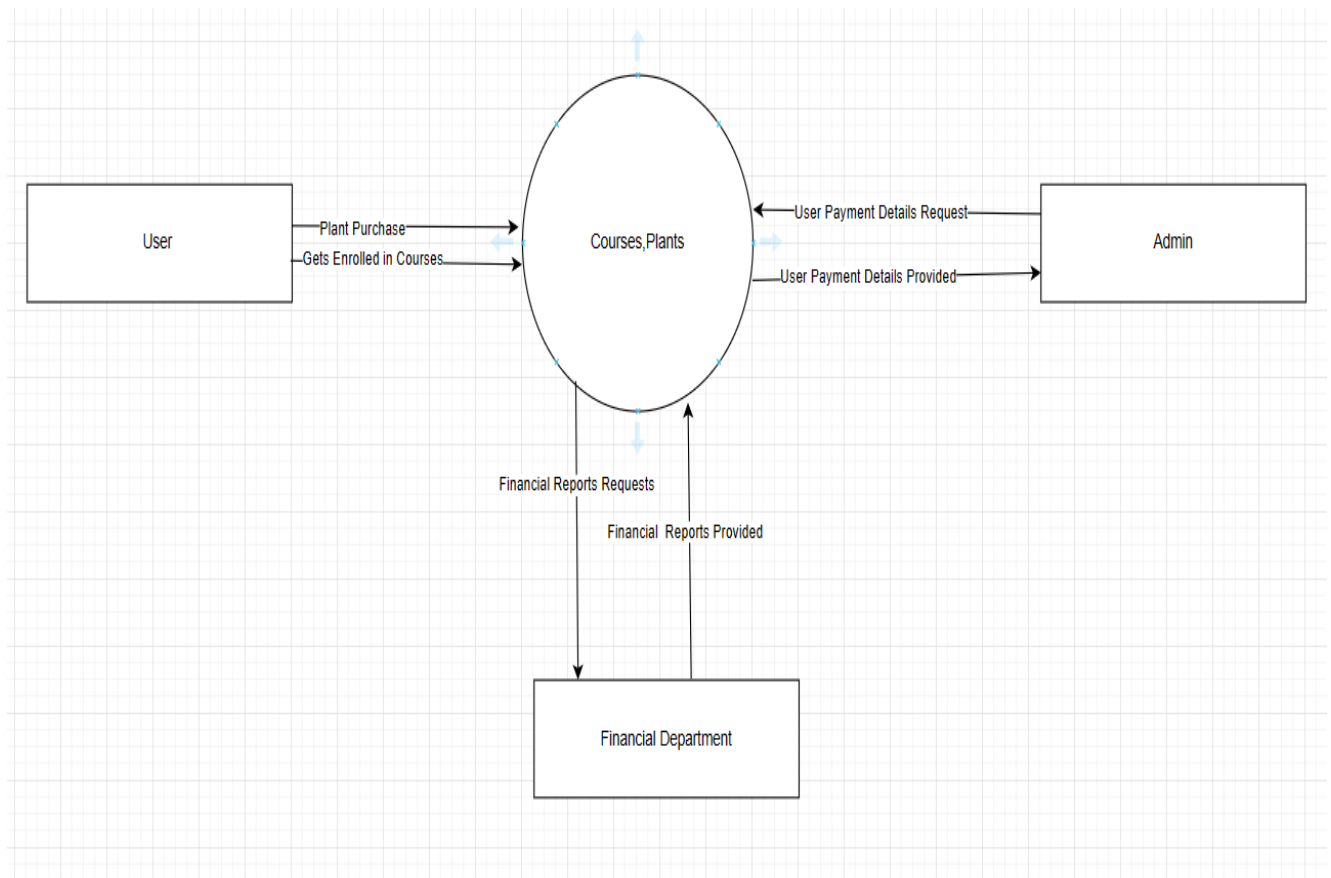


Figure 11 level 0 (make payment)

### 5.1.2 Level 1 Data Flow Diagram

A level 1 data flow diagram also known as the first level data flow diagram provides a more detailed view of the information system in comparison to Context Diagram or Level 0 data flow diagram. It divides main process of the system into different sub-processes, providing a closer look at how the system function actually works.

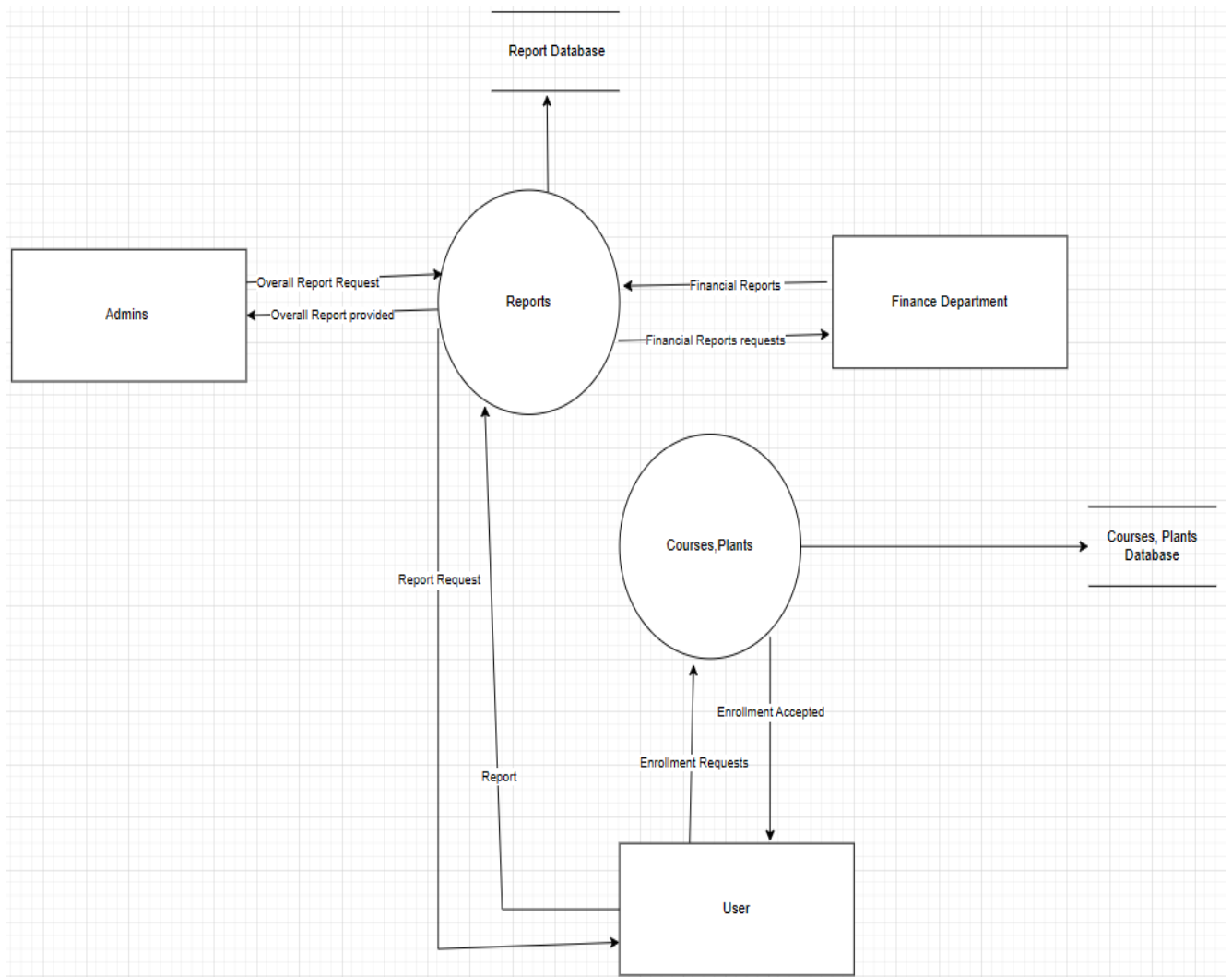


Figure 12 level 1 (make payment)

### 5.1.3 Level 2 Data Flow Diagram

A level 2 data flow diagram is a more detailed version of a Level 1 data flow diagram. It divides the processes in level 1 data flow diagram into smaller sub-processes and contains data flows within the smaller sub-processes which provides even more granular view of the system. The level of detailed work in level 2 data flow diagram is particularly

useful for inner workings of the system and is also useful in identifying the areas of improvement.

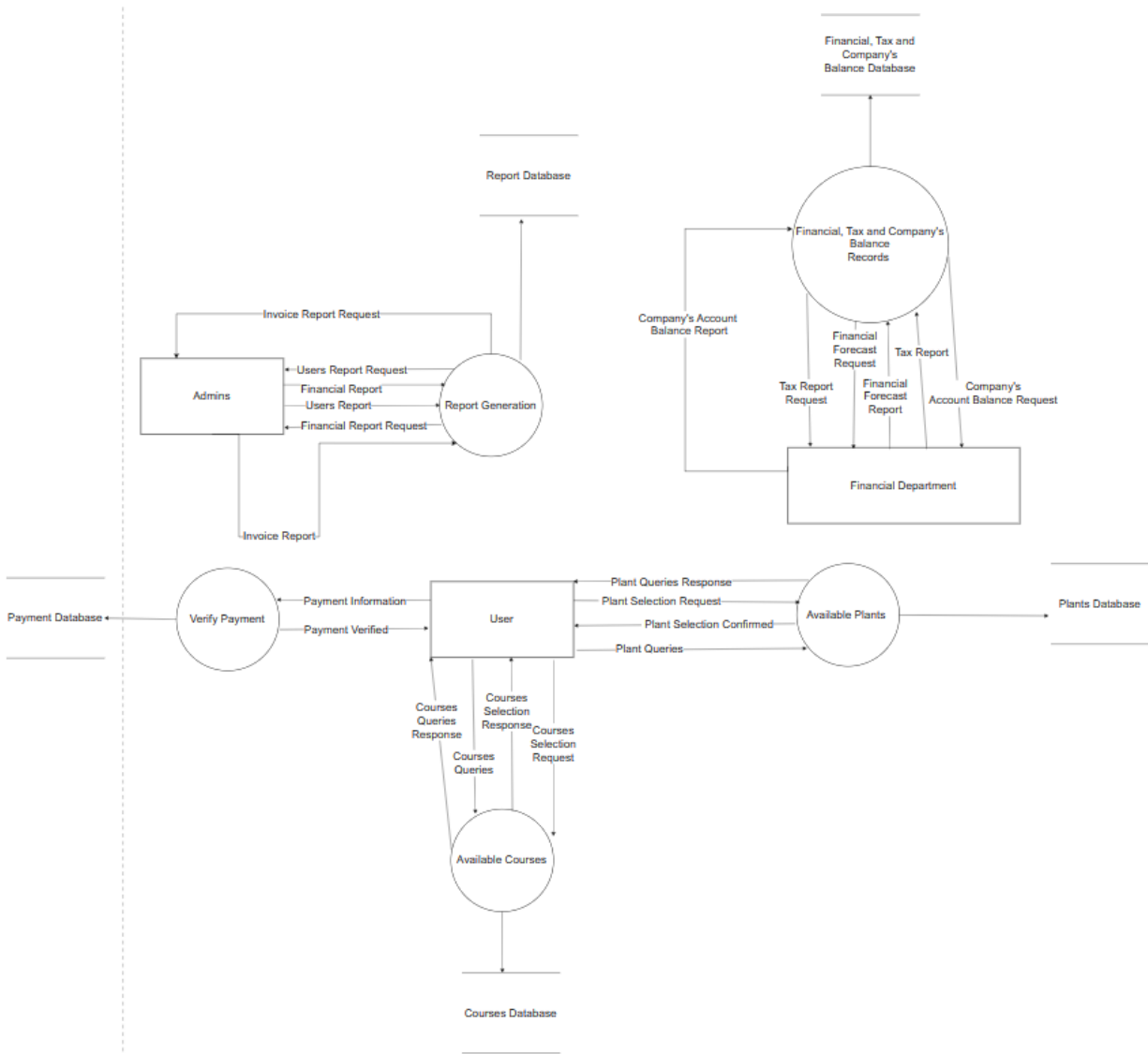


Figure 13 level 2 (make payment)

### 5.1.4 Structure Chart

A visual representation of the modules, parts of a system or static relationships of functions is known as structure chart. It's a diagram which is hierarchical and is used in system analysis and structured programming to depict the breakdown of a system into its



smallest manageable parts. Structure charts are commonly used in systems design and software development to plan and visualize complex systems.

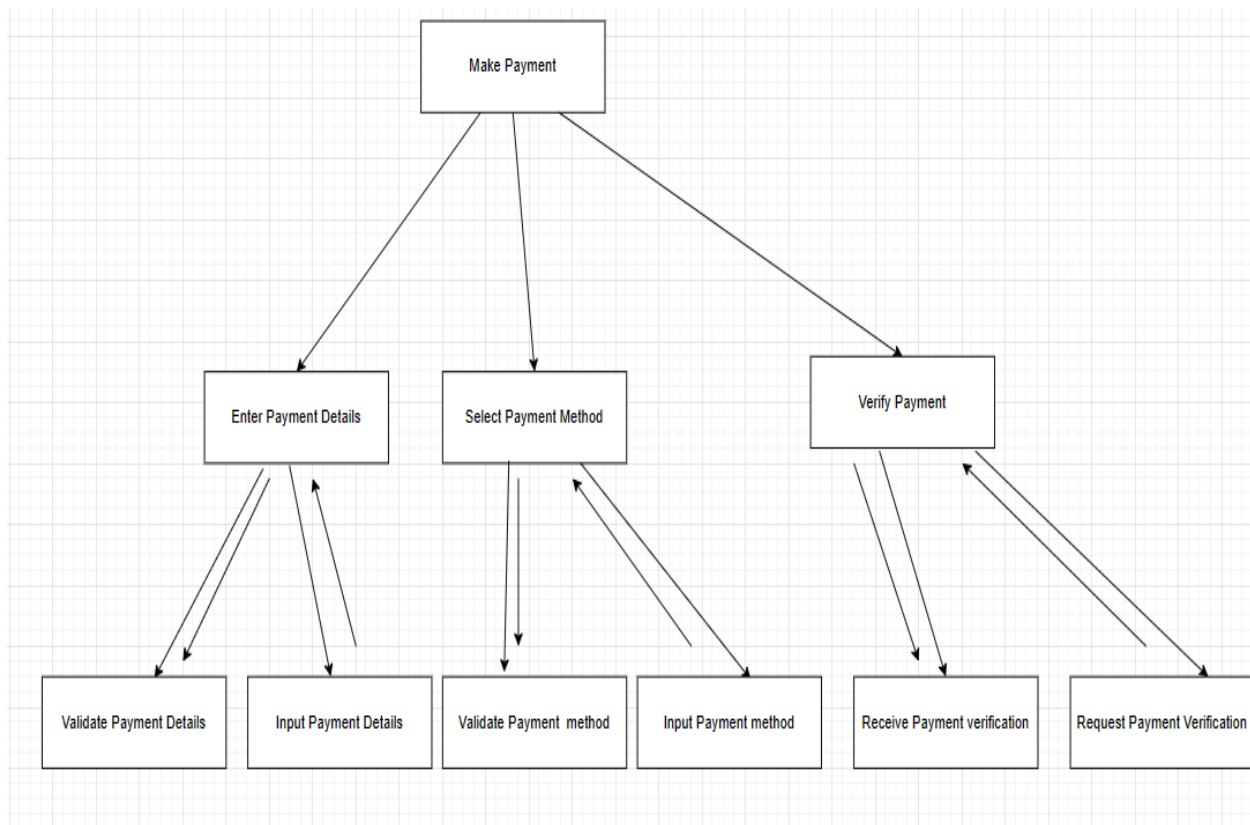


Figure 14 Structure chart (make payment)

### 5.1.5 Module Specification

Name	Purpose
Make Payment	The purpose of make payment is to facilitate transactions for the services provided by the Botanical Institute.
Pseudocode	Do Input paymentmethod, paymentdetails PaymentStatus = Verify Payment(paymentmethod, paymentdetails) If payment == successful:

	Print "Payment Successful" Else: Print "Payment Failed.Try Again"
Input Parameters	Payment Method, Payment Details, Payment Amount
Output	User Payment
Global Variables	Currency, Tax Rate, Transaction Fee
Local Variables	Paymentmethod, PaymentDetails, Paymentamount, Paymentstatus
Calls	selectPaymentmethod(), enterPaymentdetails(),verifyPayment()

Table 2 module specification (make payment)

## 5.2 Purchase Plant(Aditya Shah)

### Environment Model Specification

#### 5.2.1 Context Level Diagram

The context diagram is a crucial tool for delineating the boundaries and linkages inside a modeled system. It enumerates every entity—internal and external—as well as their interactions with the system. Often referred to as a 0-level data flow diagram, it aids in defining the parameters of a software system and the data flow details. The diagram's representation of the software system as a single process provides a comprehensive knowledge of its functionality.

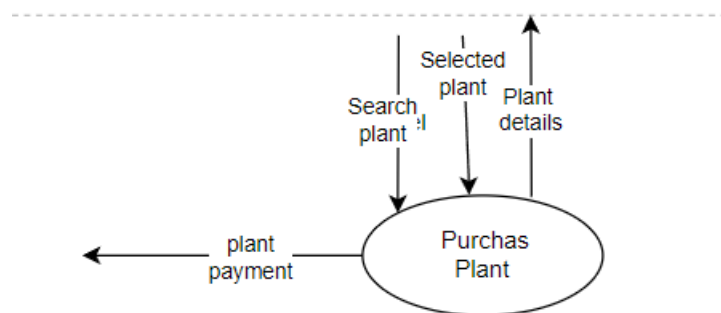


Figure 15 level0(plant purchase)

### Internal Model Specification

#### 5.2.2 Level 1 DFD

At Level-1, a data flow diagram (DFD)'s context diagram is separated into multiple bubbles or processes. Determining the functions of the system and breaking down the high-level 0-level DFD process into smaller, more manageable sub-processes are the primary objectives at this point. At a later phase, such DFD Level 1. Smaller processes are further grouped into a context diagram process node. This expansion requires extra data stores and streams to link the processes.

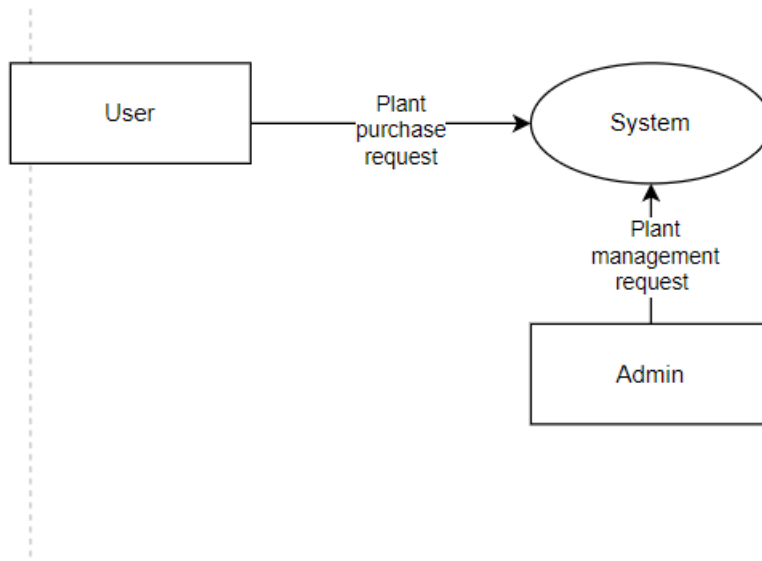


Figure 16 level1(plant purchase)

### 5.2.3 Level 2 DfD

The 2-level DFD elaborates on the outline of certain system components included in the 1-level DFD. This strict methodology makes careful planning and scheduling possible, as well as the diligent tracking of important system operations. Level 2 DFDs are not always utilized, despite the fact that they divide processes into more specialized sub-processes.

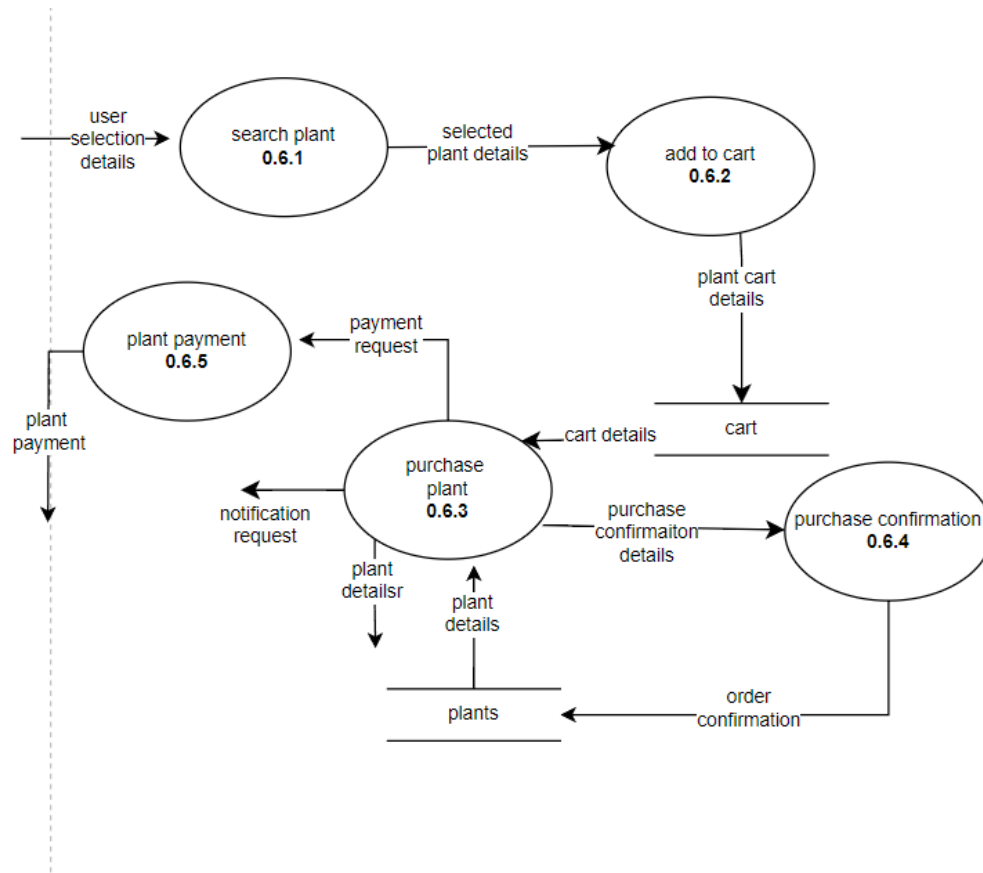


Figure 17 level2 (plant purchase)

## 5.2.4 Structure Chart

It is frequently used in software engineering to display the hierarchy of parts or modules and their connections. The primary system is at the top of the hierarchical structure, with its sub-modules branching out below it. It supports system design and development by assisting in the knowledge of a system's structure and constituent parts.

## 5.2.5 Module Specification for Plant Purchase:

### Search Plant Module:

Purpose: To allow users to search for plants based on their preferences.

Pseudocode:

INPUT: User Selection Details

IF User Selection Details are valid

    THEN Perform Plant Search

ELSE

Set Search Error

Input Parameters: User Selection Details

Output Parameters: Plant Details

Global Variables: Plant Data Store

Local Variables: User Selection Details, Plant Details

Calls: None

Called By: Add to Cart Module

**Add to Cart Module:**

Purpose: To add selected plants to the user's cart.

Pseudocode:

INPUT: Selected Plant Details

IF Selected Plant Details are valid

THEN Add to Cart

ELSE

Set Cart Addition Error

Input Parameters: Selected Plant Details

Output Parameters: Cart Details

Global Variables: Cart Data Store

Local Variables: Selected Plant Details, Cart Details

Calls: None

Called By: Purchase Plant Module

**Purchase Plant Module:**

Purpose: To facilitate the purchase of plants from the user's cart.

Pseudocode:

INPUT: Cart Details

IF Cart Details are valid

THEN Initiate Purchase

ELSE

Set Purchase Error

Input Parameters: Cart Details

Output Parameters: Purchase Confirmation

Global Variables: Purchase Data Store

Local Variables: Cart Details, Purchase Confirmation

Calls: None

Called By: Purchase Confirmation Module

**Purchase Confirmation Module:**

Purpose: To confirm the successful purchase of plants and store payment details.

Pseudocode:

INPUT: Purchase Confirmation

IF Purchase Confirmation is valid

    THEN Store Payment Details

ELSE

    Set Confirmation Error

Input Parameters: Purchase Confirmation

Output Parameters: Payment Details

Global Variables: Payment Data Store

Local Variables: Purchase Confirmation, Payment Details

Calls: None

Called By: Data Store Module

**Data Store Module:**

Purpose: To store relevant data after plant purchase, including payment details.

Pseudocode:

INPUT: Plant Payment Details

IF Plant Payment Details are valid

    THEN Store Data

ELSE

    Set Data Store Error

Input Parameters: Plant Payment Details

Output Parameters: Stored Data

Global Variables: Data Store

Local Variables: Plant Payment Details, Stored Data

Calls: None

Called By: None



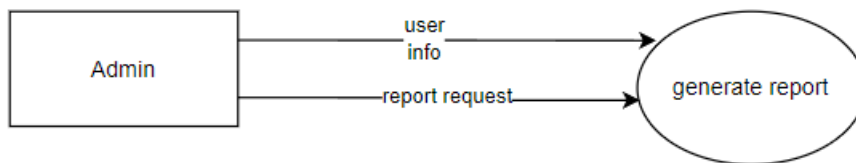
### 5.3 Report Preparation(Roshani Rauniyar)

In Sparkle Botanical System, the "Report Preparation" function is critical for administrative chores. It gives admins the ability to create thorough financial, employees, and user reports. This feature provides the administrator with information on the institute's financial transactions, employee performance, and user involvement.

#### Environment model specification

##### 5.3.1 Context level diagram

DFD 0 gives an overview of the complete system. It depicts the primary entities and their interactions (Users, Admin, Experts). Users perform a variety of tasks, while the Admin controls system operations, including the critical "Report Preparation" function. Experts are critical components of information sharing. This high-level diagram lays the groundwork for comprehensive process breakdowns in later tiers of data flow diagrams, providing a complete picture of the system's architecture.



*Figure 18 Level 0( report generation)*

##### 5.3.2 Level 1

DFD 1 looks deeper into the "Report Preparation" function, dissecting it into several stages. At this level, critical procedures such as data collecting, financial report generation, employee report generation, and user report generation are introduced. It gives a more extensive view of how data moves inside the system, making it easier to create comprehensive reports.

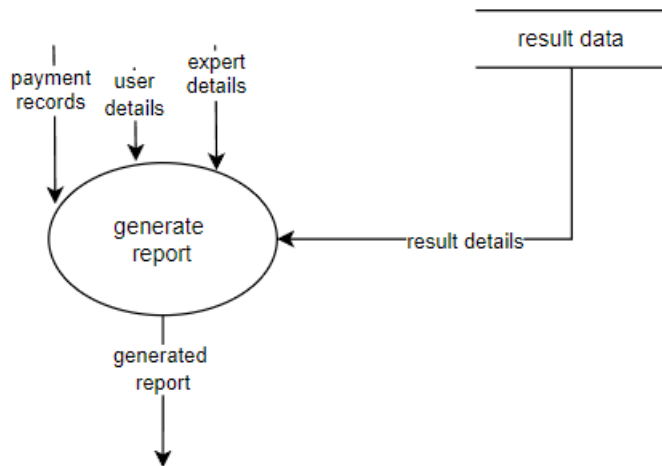


Figure 19 Level 1(report Generation)

### 5.3.3 Level 2

Within the "Report Preparation" feature, DFD 2 carefully tracks the process of acquiring user details, opinions of experts, and payment information. The stage starts with the collection of user-specific data, expert views, and payment records. This aggregated data is then sent to the individual report creation module, which generates unique user reports based on the acquired data. Following that, all individual reports are collated and sent to the general report generating process. Individual reports are integrated into the entire report creation process, ensuring a full picture of financial, personnel, and user-related data. This level gives a full view of data collection, processing, and aggregation in order to provide complete and cohesive overall reports.

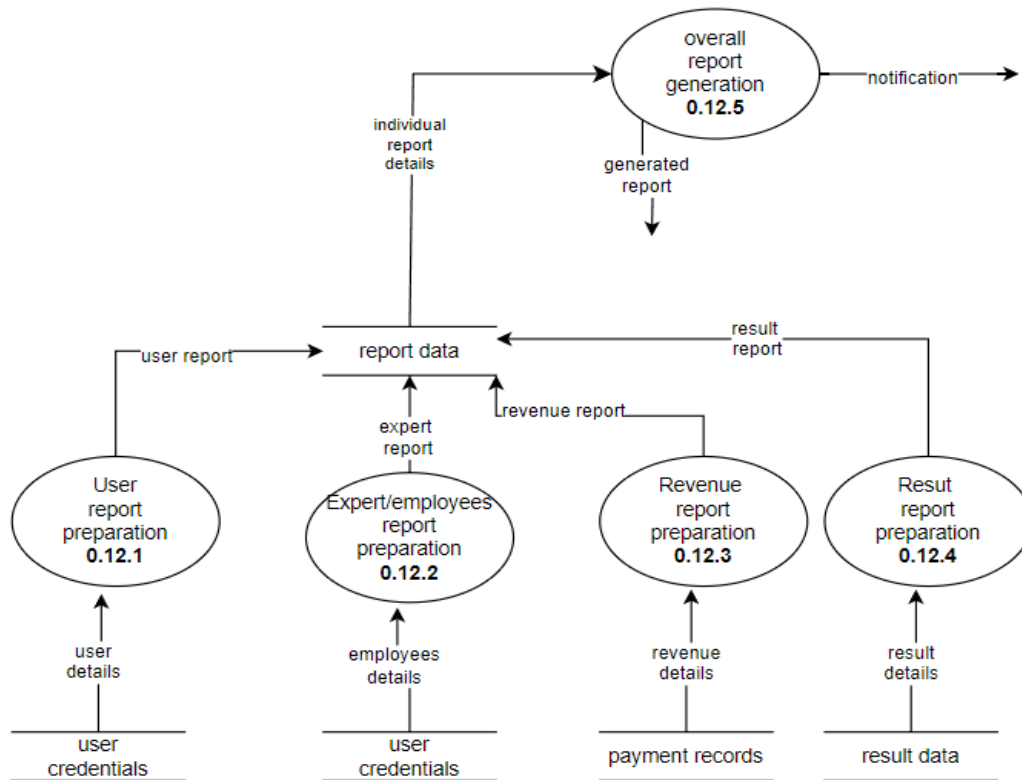


Figure 20 Level 2(report Generation)

### 5.3.4 Structure Chart

The Structure Chart is an important component that uses relationships to depict the data structure and flow inside a system. It exposes the system's hierarchical linkages, exhibiting interconnections between processes, functions, data storage, and the flow of both data and control. A Structure Chart, in the context of "Report Preparation" for McGregor Institute, provides a visual depiction of the logical processes involved in preparing reports. It explains the orderly flow of data by displaying the system hierarchy, which includes all critical activities. The procedure begins with the administrator gathering user information, expert insights, and payment information, which is then put into individual reports. Individual reports are then consolidated into an overall report, which provides a full picture of financial, personnel, and user information. The Structure Chart is a visual assistance for reviewing and improving the efficiency and effectiveness of the McGregor Institute's "Report Preparation" function. (Chart?, 2022)

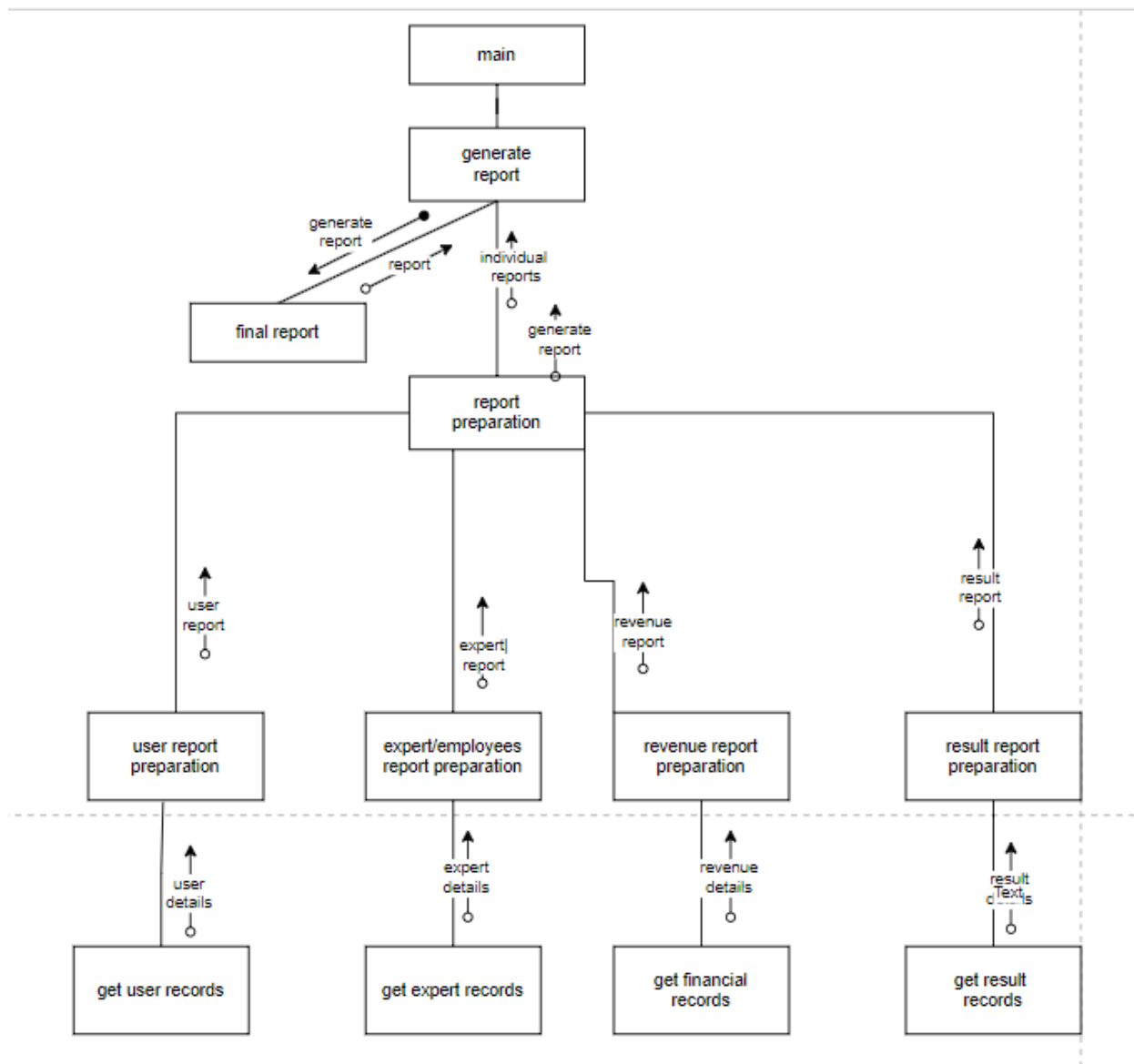


Figure 21 Structure chart(report generation)

### 5.3.5 Module Specification

The "Report Preparation" module assembles detailed reports by aggregating financial, employee, and user data. Inputs include user preferences and expert insights, with outputs comprising personalized user reports and updated financial records. Functions involve data aggregation, report customization, and expert insight integration. Dependencies include reliable data sources and adherence to predefined criteria. Constraints encompass ethical payment information handling. Rigorous testing ensures

accuracy, and seamless integration with other modules guarantees effective system performance. This concise specification outlines the key functionalities and requirements for the "Report Preparation" module.

#### 5.1.5.1 Module Specs for Level 2 DFD

##### User report preparation module specification:

Name	User Report Preparation
Purpose	To compile and generate a detailed report based on user details.
Pseudocode	<pre> INPUT User Details FROM User credentials IF User Details is not null     THEN         Perform User Report Preparation     ELSE         Set Report Generation Error </pre>
Input Parameters:	User Details (User_ID, name, address, etc.)
Output Parameters:	User Report
Global variables:	DB
Local variables:	User Details, User Report
Calls:	None
Calls by:	Overall report generation

*Table 3 user report preparation*

##### Expert report preparation module specification:

Name	Expert Report Preparation
Purpose	To compile and generate a detailed report based on expert details.
Pseudocode	INPUT Expert Details FROM user credentials

	IF Expert Details is not null THEN Perform Expert Report Preparation ELSE Set Report Generation Error
Input Parameters:	Expert Details (expert_ID, name, address, etc.)
Output Parameters:	expert Report
Global variables:	DB
Local variables:	Expert Details, Expert Report
Calls:	None
Calls by:	Overall report generation

*Table 4 expert report preparation*

#### **Financial report preparation module specification:**

Name	Financial Report Preparation
Purpose	To compile and generate a detailed financial based on payment details.
Pseudocode	INPUT Payment Details FROM Payment records IF Payment Details is not null THEN Perform Financial Report Preparation ELSE Set Report Generation Error
Input Parameters:	Payment Details (transaction_ID, amount, payment_status, etc.)
Output Parameters:	Financial Report
Global variables:	DB
Local variables:	Payment Details, Financial Report

Calls:	None
Calls by:	Overall report generation

*Table 5 financial report preparation*

**Exam report preparation module specification:**

Name	Exam Report Preparation
Purpose	To compile and generate a detailed report based on exam details.
Pseudocode	<pre> INPUT Exam Details FROM Exam Data Store IF Exam Details is not null     THEN         Perform Exam Report Preparation     ELSE         Set Report Generation Error </pre>
Input Parameters:	Exam Details (exam_ID, results, completion_status, etc.)
Output Parameters:	Exam Report
Global variables:	DB
Local variables:	Exam Details, Exam Report
Calls:	None
Calls by:	Overall report generation

*Table 6 Exam report preparation*

**Overall report generation module specification:**

Name	Overall Report Generation
Purpose	To compile and generate an overall report based on individual reports.
Pseudocode	INPUT:

	User Report ,Expert Report , Financial Report and Exam Report FROM report data store IF User Report is not null AND Expert Report is not null AND Financial Report is not null AND Exam Report is not null THEN Perform Overall Report Generation ELSE Set Report Generation Error
Input Parameters:	User Report, Expert Report, Financial Report, Exam Report
Output Parameters:	Overall Report
Global variables:	DB
Local variables:	User Report, Expert Report, Financial Report, Exam Report, Overall Report
Calls:	Notification Module
Calls by:	Notification Output

Table 7 Overall report preparation

#### Notification Module Specification:

Name	Notification
Purpose	To notify users about the completion of report generation.
Pseudocode	IF Overall Report Generation is successful THEN Notify User ELSE Report Generation Error
Input Parameters	Overall Report
Output Parameters	Notification
Global Variables	DB



Local Variables	Notification
Calls	None
Called By	Overall Report Generation

*Table 8 Notification*

### **5.3.6 Overall Description**

The Report Generation process within the 'Sparkle Botanical System' is structured across three levels, offering a comprehensive view of the synthesis of reports from diverse data channels. At DFD Level 0, the process encompasses the entire workflow, receiving inputs from users, experts, payments, and results. These inputs are directed to specific sub processes at DFD Level 1: 'User Report Preparation,' 'Expert Report Preparation,' 'Financial Report Preparation,' and 'Result Report Preparation.' Each sub process, detailed at DFD Level 2, engages in specific tasks such as data collection, and contributing to the creation of individual reports. The synthesized reports from each sub process converge at the 'Overall Report Generation,' providing a consolidated and holistic overview. The Structure Chart visually depicts the logic within the process, showcasing the hierarchical relationships among modules and functions. Module Specification further details the functionality of each module, emphasizing the modular and structured nature of the report generation process. This structured approach ensures a streamlined and efficient synthesis of comprehensive reports within the 'Sparkle Botanical System.'

## 5.4 Join the program(Aditi Basnet)

### Join the program

The institute provides Undergraduate and Graduate courses to interested individuals. There are both paid and unpaid courses where users can enroll.

#### 5.4.1 Context Level Diagram

The context diagram is a crucial tool for defining the boundaries and relationships of a modeled system. It identifies the internal and external entities and their interplay with the system. Often referred to as a 0-level data flow diagram, it serves to clarify the scope of a software system and the data flow between it and external devices. The diagram represents the entire software system as a single process, providing a comprehensive view of its function.

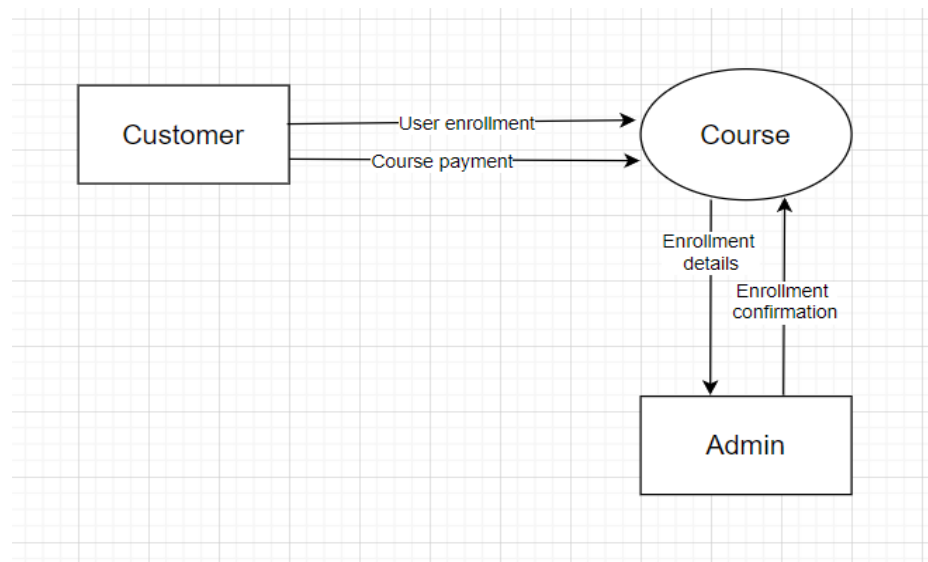


Figure 22 level 0( Join the program)

#### 5.4.2 Level 1 DFD

At the Level-1 of a data flow diagram (DFD), the context diagram is segmented into various bubbles or processes. This stage focuses on identifying the primary functions of the system, and breaking down the high-level process of the 0-level DFD into smaller sub-processes. In a subsequent level, like Level 1 DFD, a process node from the context diagram is further divided into sub-processes. This expansion requires additional data

streams and data stores to link the processes. For instance, in the context of a hotel reservation system, this may involve incorporating room selection and inquiry processes, as well as data warehouses.

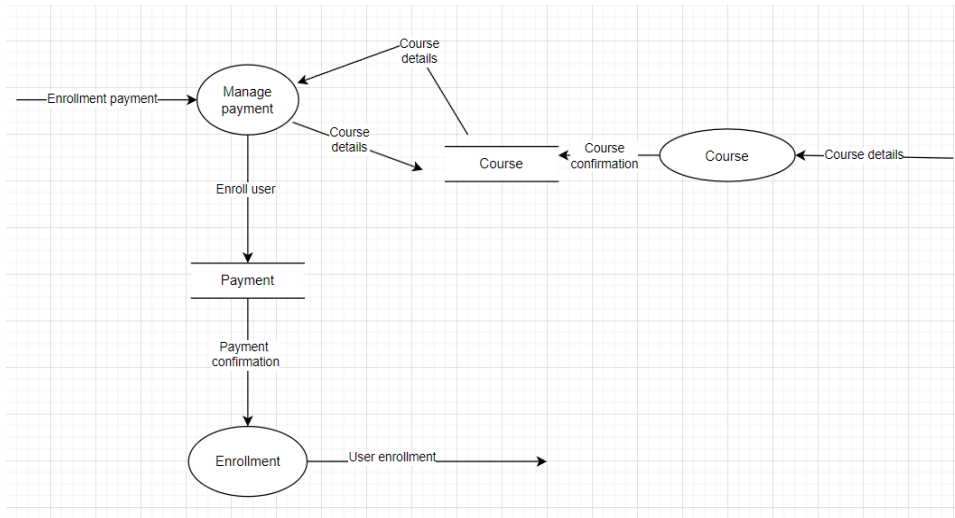


Figure 23 level 1( Join the program)

### 5.4.3 Level 2 DFD

The 2-level DFD delves deeper into specific aspects of the system that were outlined in the 1-level DFD. This methodical approach allows for careful planning and detailed tracking of important processes within the system. While level 2 DFDs break down processes into more specific sub-processes, it is rare for DFDs to progress beyond level 3 due to the level of detail already provided. Typically, level 3 data flow diagrams are considered sufficiently detailed and do not require further breakdown.

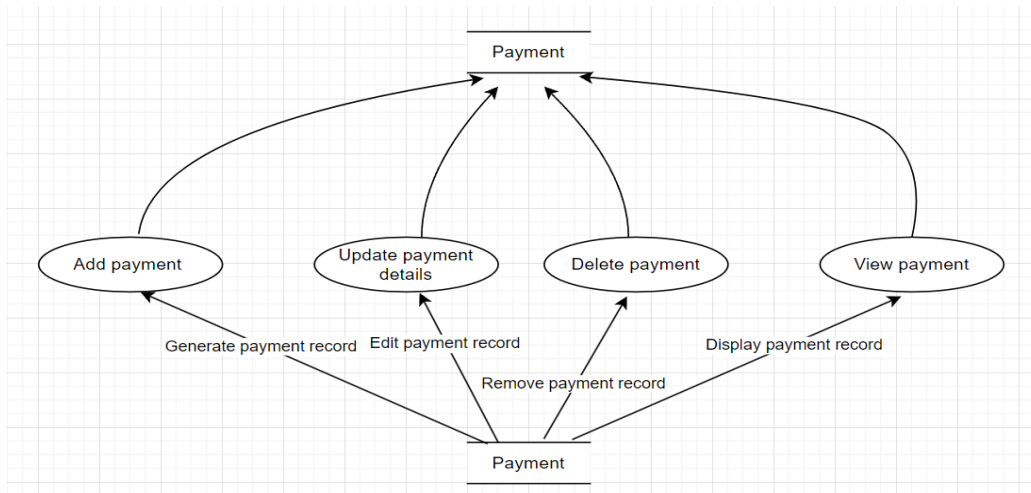


Figure 24 level 2( Join the program)

#### 5.4.4 Structure Chart

In the realm of software engineering and organizational theory, a Structure Chart (SC) is a valuable tool that breaks down the configuration system into its most manageable components. Essentially, it is a tree structure that arranges program modules in a way that facilitates visualization of their interrelationships. The chart is comprised of boxes, each of which represents a module and contains its name.

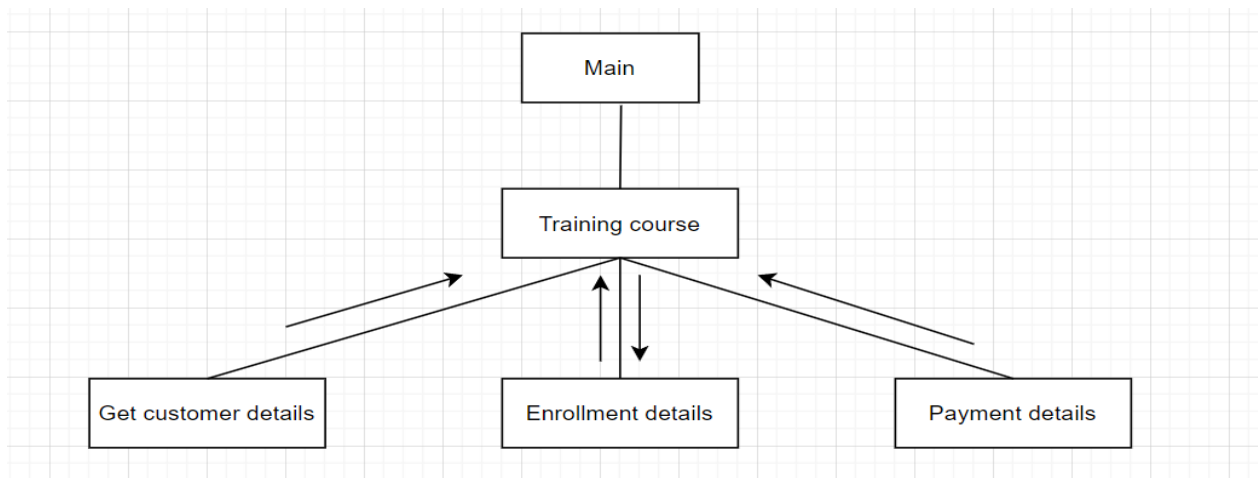


Figure 25 Structure Chart( Join the program)

### 5.4.5 Module Specification

<b>Name</b>	Join the program
<b>Purpose</b>	<b>This module is for the enrollment of users who want to take short term courses or certification courses; both paid and unpaid course options are available.</b>
<b>Pseudocode</b>	<b>DO</b> var user_details = DB.get_user_details() var payment_details=DB.get_payment_details() var enrollment_details=DB.get_enrollment_details()  var user_id = user_details['user_details']  var User_report= generate_report(user_details,payment_details, enrollment_details)  Display(user_report)
<b>Input parameters</b>	<b>User details, payment details, enrollment details</b>
<b>Output</b>	<b>User_joined_program</b>
<b>Global variables</b>	<b>DB</b>
<b>Local variables</b>	<b>User_details, payment_details, enrollment_details and user_report</b>
<b>Calls</b>	<b>Get User details, payment details, enrollment details and generate User report</b>

Table 9 table of join the program

## 5.5 Take Certification Exam (Lata Kumari Thakurathi)

Developing a thorough certification test requires a methodical strategy to precisely assess candidates' knowledge and abilities. Exam objectives are defined clearly at the outset of the process, along with the purpose and target audience.

### Environment Model Specification

#### 5.5.1 Context Level Diagram

The context level diagram is an essential tool for outlining the bounds and connections inside a modeled system. It lists all of the entities, both internal and external, and how they interact with the system. Frequently called a 0-level data flow diagram, it helps to make clear the boundaries of a software system and the data flow information. The diagram offers a thorough understanding of the functionality of the software system by representing it as a single process.

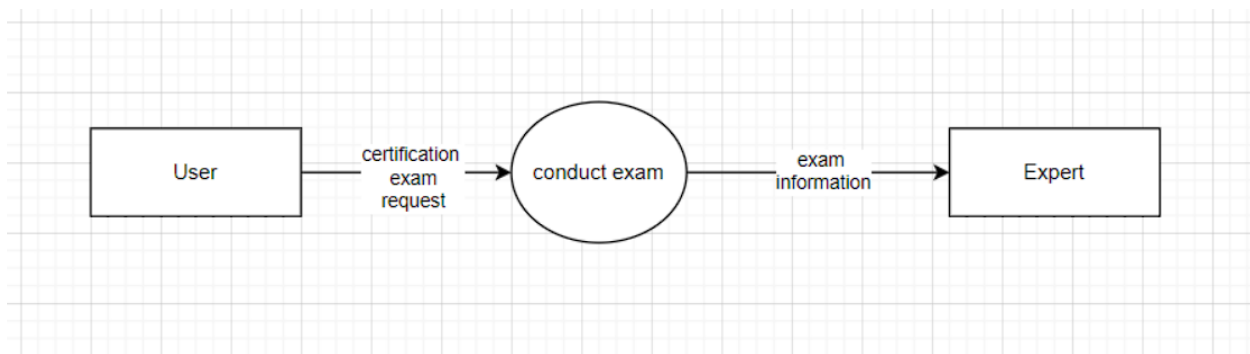


Figure 26 level 0(Certification exam)

### Internal Model Specification

#### 5.5.2 Level 1 DFD

The context diagram of a data flow diagram (DFD) is divided into several bubbles or processes at Level-1. At this stage, the main goals are to determine the system's functions and decompose the high-level 0-level DFD process into more manageable sub-processes. In a later stage, such as Level 1 DFD. A context diagram process node is further subdivided into smaller processes. To connect the processes, this expansion calls for more data stores and streams.

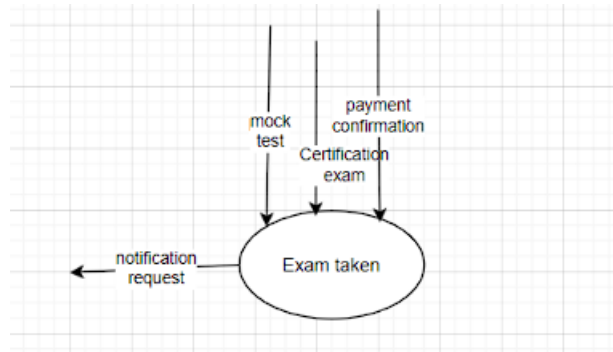


Figure 27 level 1 (Certification exam)

### 5.5.3 Level 2 DFD

The 1-level DFDs outline of particular system components is expanded upon in the 2-level DFD. This rigorous approach enables meticulous tracking of critical system activities as well as cautious planning and scheduling. Although level 2 DFDs segment processes into more focused sub-processes, they are not always used.

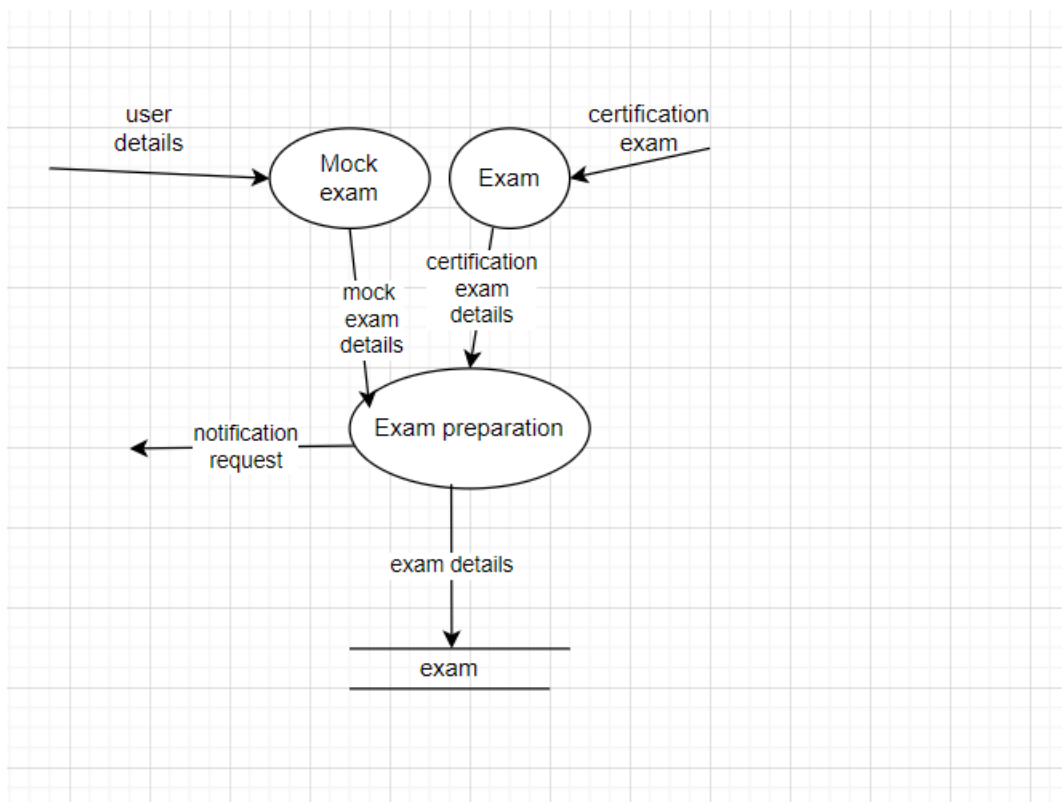


Figure 28 level 2 (Certification exam)

### 5.5.4 Structure Chart

A Structure Chart (SC), which divides the configuration system into its most manageable parts, is an invaluable tool in the fields of organizational theory and software engineering. It is essentially a tree structure that organizes program components. It makes their relationships easier to visualize. The chart consists of boxes, each with a name and a representation of a module.

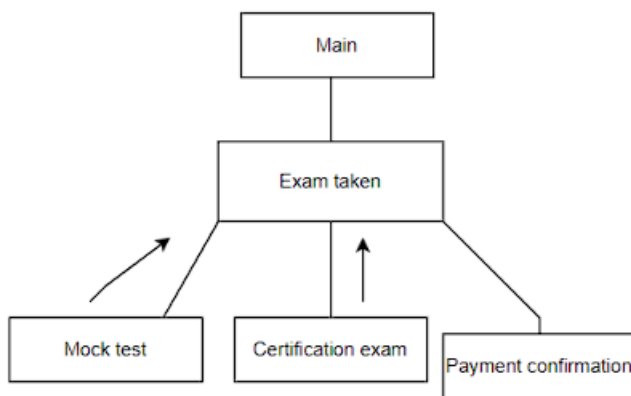


Figure 29 structure chart (Certification exam)

Name	Exam Preparation
Purpose	To facilitate the preparation and administration of certification exams.
Pseudocode	FETCH User Details FROM User Details Data Store IF User Details exist THEN Prepare Mock Exam ELSE Report Preparation Error FETCH Certificate Exam FROM Certificate Exam Data Store IF Certificate Exam exists



	THEN Enter Exam ELSE Report Preparation Error FETCH Exam Details FROM Exam Data Store
Input Parameters	User Details (User_ID, name, address, etc.)
Output	Exam Details
Global variables	DB
Local variables	User Details, Certificate Exam, Exam Details
Call	Notification Module
Called by	Notification Output

*Table 10 take certificate exam*

## 6 Summary

The systematic development of the 'Sparkle Botanical System' underscores our dedication to meeting the objectives of the McGregor Institute of Botanical Training. We have seamlessly applied crucial methodologies, including DFD, Structure Chart, Data Dictionary, Module Specification, Process Modules, and ER-Diagram, positioning the system for effective implementation in practical settings. Each group member's collaborative efforts played a pivotal role in surmounting challenges, with group meetings serving as platforms for insightful discussions and innovative problem-solving. Our comprehensive research, spanning various sources such as books, websites, and journals, has significantly enriched our understanding of vital software engineering concepts.

In the comprehensive group task, we engaged in environmental model specification, internal model specification, data dictionary, and design specification, establishing a robust foundational layer for the 'Sparkle Botanical System.' These designs were meticulously refined in individual tasks, where each group member focused on environmental model specification, internal model specification, and design specifications for specific functions.

The iterative refinement process was supported by ongoing meetings and discussions with our esteemed module lecturer, ensuring our progress aligned strategically with overarching project objectives. This collective effort has not only led to the successful completion of the 'Sparkle Botanical System' but has also deepened our understanding of critical software engineering concepts, particularly in DFD, data dictionary, structure chart, E-R Diagram module specs, and process specs.

While the current system design impeccably meets group requirements, its inherent flexibility allows for seamless updates to cater to user-specific needs in practical scenarios. The triumphant conclusion of the project has undoubtedly honed key software engineering skills, encompassing requirement analysis, project management, structured software engineering, and adept problem-solving. The collaborative approach of our small group, combined with effective project management and unwavering adherence to deadlines, has culminated in the resounding success of the 'Sparkle Botanical System.'

## References

Baluch, A., 2023. *What Is A Project Charter? Everything You Need To Know*. [Online] Available at: <https://www.forbes.com/advisor/business/project-charter/> [Accessed 29 December 2023].

Chart?, W. I. t. P. o. a. S., 2022. *What Is the Purpose of a Structure Chart?*. [Online] Available at: <https://smallbusiness.chron.com/purpose-structure-chart-50946.html> [Accessed 4 January 2024].

Chia, A., 2023. *Data Dictionaries: A Comprehensive Guide*. [Online] Available at: [https://www.splunk.com/en\\_us/blog/learn/data-dictionary.html](https://www.splunk.com/en_us/blog/learn/data-dictionary.html) [Accessed 4 January 2024].

Chi, C., 2023. *A Beginner's Guide to Data Flow Diagrams*. [Online] Available at: <https://blog.hubspot.com/marketing/data-flow-diagram> [Accessed 4 January 2024].

Geeksforgeeks, 2023. *Functional vs Non Functional Requirements*. [Online] Available at: <https://www.geeksforgeeks.org/functional-vs-non-functional-requirements/> [Accessed 30 December 2023].

geeksforgeeks, 2023. *Levels in data flow diagrams (DFDs)*. [Online] Available at: [https://l.instagram.com/?u=https%3A%2F%2Fwww.geeksforgeeks.org%2Flevels-in-data-flow-diagrams-dfd%2F&e=AT3RIU-C1plqdVD2dhBBQEWDeJhbpHPb0bqYPWaL6zuAZrtOpv0PCx8xawGifyR41qOl5jZZoyEHh6oJ4dWXDvqC\\_v9k\\_34AS54TDtF8Fq19eYYwBhgPUeTsvfwPWUwA8VhdgE](https://l.instagram.com/?u=https%3A%2F%2Fwww.geeksforgeeks.org%2Flevels-in-data-flow-diagrams-dfd%2F&e=AT3RIU-C1plqdVD2dhBBQEWDeJhbpHPb0bqYPWaL6zuAZrtOpv0PCx8xawGifyR41qOl5jZZoyEHh6oJ4dWXDvqC_v9k_34AS54TDtF8Fq19eYYwBhgPUeTsvfwPWUwA8VhdgE) [Accessed 5 January 2024].

Lane, C., 2023. *What Is a Software Requirements Specification (SRS) Document?*. [Online] Available at: [https://www.perforce.com/blog/alm/how-write-software-requirements-specification-srs-document#:~:text=A%20software%20requirements%20specification%20\(SRS\)%20is%20a%20document%20that%20describes,will%20be%20expected%20to%20perform.](https://www.perforce.com/blog/alm/how-write-software-requirements-specification-srs-document#:~:text=A%20software%20requirements%20specification%20(SRS)%20is%20a%20document%20that%20describes,will%20be%20expected%20to%20perform.) [Accessed 30 December 2023].

MAR31, 2023. *Internal models approach: model requirements*. [Online] Available at: [https://www.bis.org/basel\\_framework/chapter/MAR/31.htm?inforce=20220101&published=20191215](https://www.bis.org/basel_framework/chapter/MAR/31.htm?inforce=20220101&published=20191215) [Accessed 4 January 2024].

S, R. A., 2023. *ER Diagrams in DBMS: Entity Relationship Diagram Model*. [Online] Available at: <https://www.simplilearn.com/tutorials/sql-tutorial/er-diagram-in-dbms> [Accessed 4 January 2024].

