Chapter Three

Analysis Design

3.1. Introduction

As mentioned in chapter two System analysis and design includes all activities, which help the transformation of requirement specification into implementation. Requirement specifications specify all functional and non-functional expectations from the system. These requirement specifications come in the shape of human readable and understandable documents.

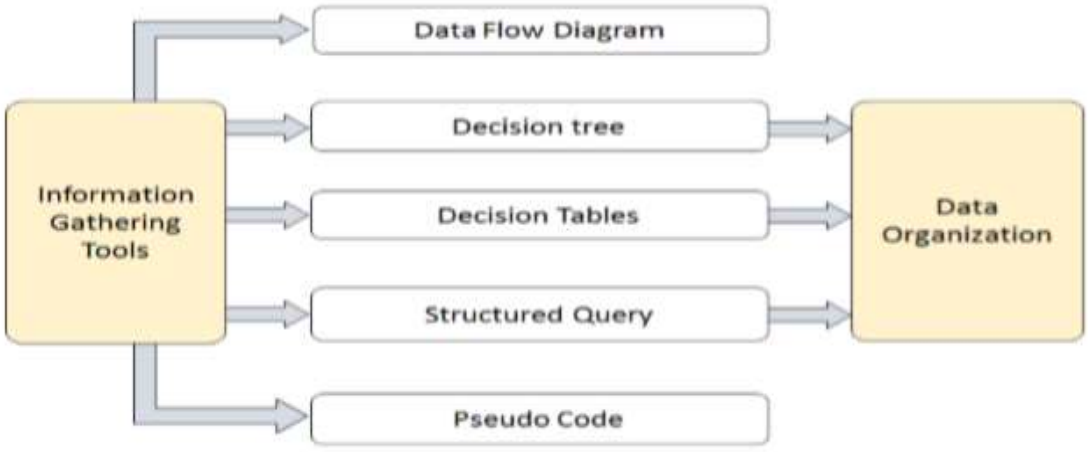
System analysis and design is the intermediate stage, which helps human-readable requirements to be transformed into actual code in implementation phase.

كما ذكر في الفصل الثاني تحليل وتصميم النظام يشمل جميع الأنشطة، التي تساعد على التحويل من مواصفات المتطلبات الى شيء قابل للتنفيذ .مواصفات المتطلبات تحدد كل التوقعات الوظيفية وغير الوظيفية من النظام .وتأتي مواصفات المتطلبات هذه في شكل وثائق قابلة للقراءة ويمكن فهمها من قبل الاشخاص.

ويعد تحليل النظام وتصميمه من المراحل الوسطية، التي تساعد على تحويل المتطلبات التي يمكن قراءتها من قبل الاشخاص إلى كود فعلي يتم تنفيذه في مرحلة التنفيذ.

Let us see few analysis and design tools used by system designers:

- **Data Flow Diagrams**
- **Data Dictionary**
- **Decision Trees**
- **Decision Tables**
- **Structured English**
- **Pseudocode**
- **Use-case**



1. Data Flow Diagrams (DFD) or Bubble Chart

It is a technique developed by Larry Constantine to express the requirements of system in a graphical form.

هي تقنية التي وضعها لاري قسطنطين للتعبير عن متطلبات النظام في شكل رسومي

A DFD also known as a "bubble chart," has the purpose of clarifying system requirements and identifying major transformations that will become programs in system design. So it is the starting point of the design phase that functionally decompose the requirements specifications down to the lowest level of detail. **A DFD consists of a series of bubbles joined by lines. The bubbles represent data transformations and the lines represent data flows in the system.**

و هي ايضا معروفة باسم "مخطط الفقاعات او الدوائر"، والغرض منها هو توضيح متطلبات النظام وتحديد التحويلات الرئيسية التي سوف تصبح برامج في تصميم النظام .لذلك هي نقطة الانطلاق لمرحلة التصميم التي تعمل وظيفيا على تجزئة مواصفات المتطلبات وصولا الى أدنى مستوى من التفاصيل .يتكون ال DFD من سلسلة من الفقاعات المرتبطة بواسطة خطوط .تمثل الفقاعات عمليات تحويل البيانات وتمثل الخطوط تدفقات البيانات في النظام.

i. Basic Elements of DFD

In the DFD, there are four symbols.

1. **A square** defines a source (originator) or destination of system data.

المربع مصدر (منشئ) أو وجهة بيانات النظام

2. **An arrow** identifies data flow- data in motion. It is a pipeline through which information flows.

يحدد السهم تدفق البيانات- البيانات في حالة الحركة. انه خط من خلاله تتدفق المعلومات

3. **A circle or a "bubble"** (some people use an oval bubble) represents a process that transforms incoming data flow(s) into outgoing data flow(s).

دائرة أو "فقاعة" (بعض الاشخاص يستخدمون فقاعة بيضاوية) تمثل عملية تحول تدفق البيانات القادمة إلى تدفق البيانات الصادرة

4. **An open rectangle** is a data store- data at rest, or a temporary repository of data.

المستطيل المفتوح هو مخزن البيانات ، أو مستودع مؤقت للبيانات

The following table shows the symbols used in designing a DFD and their significance:

Symbol Name	Symbol	Meaning
Square		Source or Destination of Data
Arrow		Data flow
Circle		Process transforming data flow
Open Rectangle		Data Store

ii. Types of DFD

DFDs are of two types: Physical DFD and Logical DFD. The following table lists the points that differentiate a physical DFD from a logical DFD.

Physical DFD	Logical DFD
It is implementation dependent. It shows which functions are performed.	It is implementation independent. It focuses only on the flow of data between processes.
It provides low level details of hardware, software, files, and people.	It explains events of systems and data required by each event.
It depicts how the current system operates and how a system will be implemented.	It shows how business operates; not how the system can be implemented.

iii. Constructing a DFD

Several rules of thumb are used in drawing DFDs.

1. Process should be named and numbered for easy reference. Each name should be representative of the process.

ينبغي تسمية وترقيم العملية لسهولة الرجوع إليها .يجب أن يكون كل اسم ممثلا للعملية.

2. The direction of flow is from top to bottom and from left to right. Data traditionally flow from the source (upper left corner) to the destination (lower right corner), although they may flow back to a source.

اتجاه التدفق من أعلى إلى أسفل ومن اليسار إلى اليمين .البيانات تتدفق عادة من المصدر (الزاوية اليسرى العليا) إلى الوجهة (الزاوية اليمنى السفلى)، وعلى الرغم من ذلك أنها ممكن ان قد تتدفق رجوعا إلى المصدر .

3. When a process is exploded into lower- level details, they are numbered.

عندما تتجزء العملية إلى تفاصيل المستوى ألادنى، يتم ترقيمهم .

4. The names of data stores, sources and destinations are written in capital letters. Process and data flow names have the first letter of each word capitalized.

تكتب أسماء مخازن البيانات ومصادرها ووجهاتها بأحرف كبيرة .أسماء العمليات وتدفق البيانات يكون الحرف الأول من كل كلمة كبير.

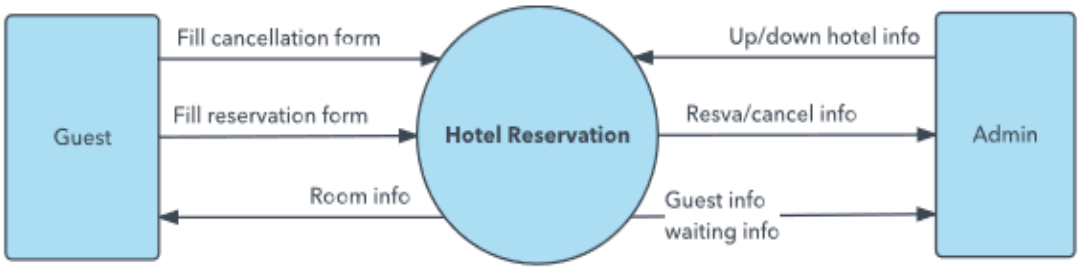
The DFD methodology is quite effective, especially when the required design is unclear and the user and the analyst need a notational language for communication. The DFD is easy to understand after a brief orientation.

The main problem however is the large number of iterations that often are required to arrive at the most accurate and complete solution.

منهجية DFD هي فعالة جدا، وخصوصا عندما يكون التصميم المطلوب غير واضح والمستخدم والمحلل بحاجة إلى لغة شائعة للاتصال .و DFD من السهل فهمها بعد التوجه الموجز. ولكن المشكلة الرئيسية هي العدد الكبير من التكرارات التي غالبا ما تكون مطلوبة للوصول إلى الحل الأكثر دقة والكامل.

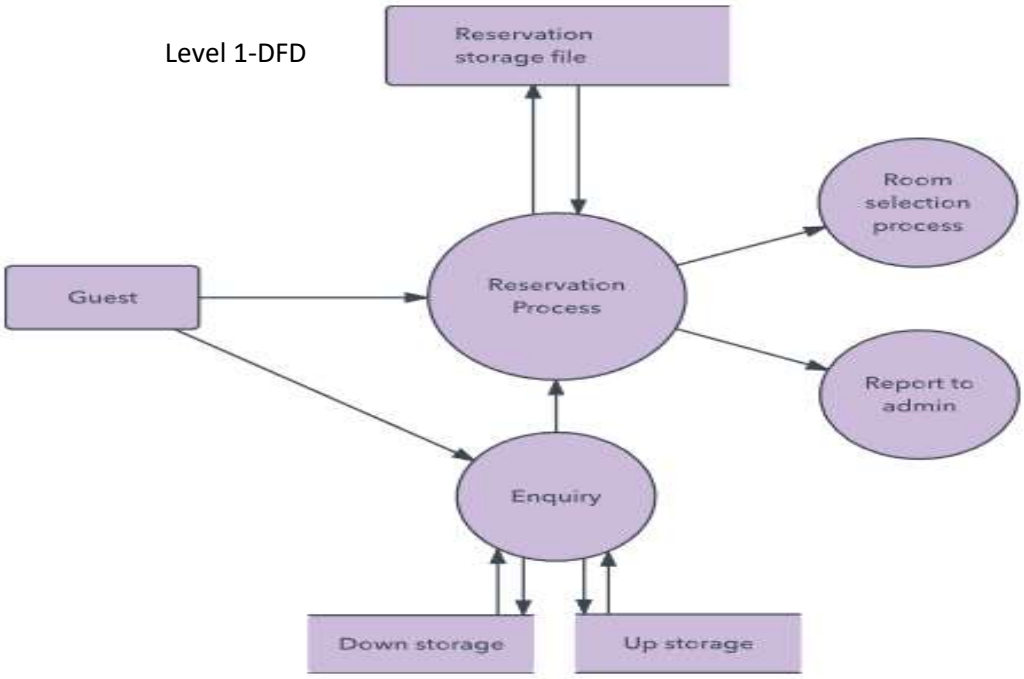
iv. Levels of DFD

- **Level 0 also known as context diagram**- Highest abstraction level DFD is known as Level 0 DFD, which depicts the entire information system as one diagram concealing all the underlying details. Level 0 DFDs are also known as context level DFDs.

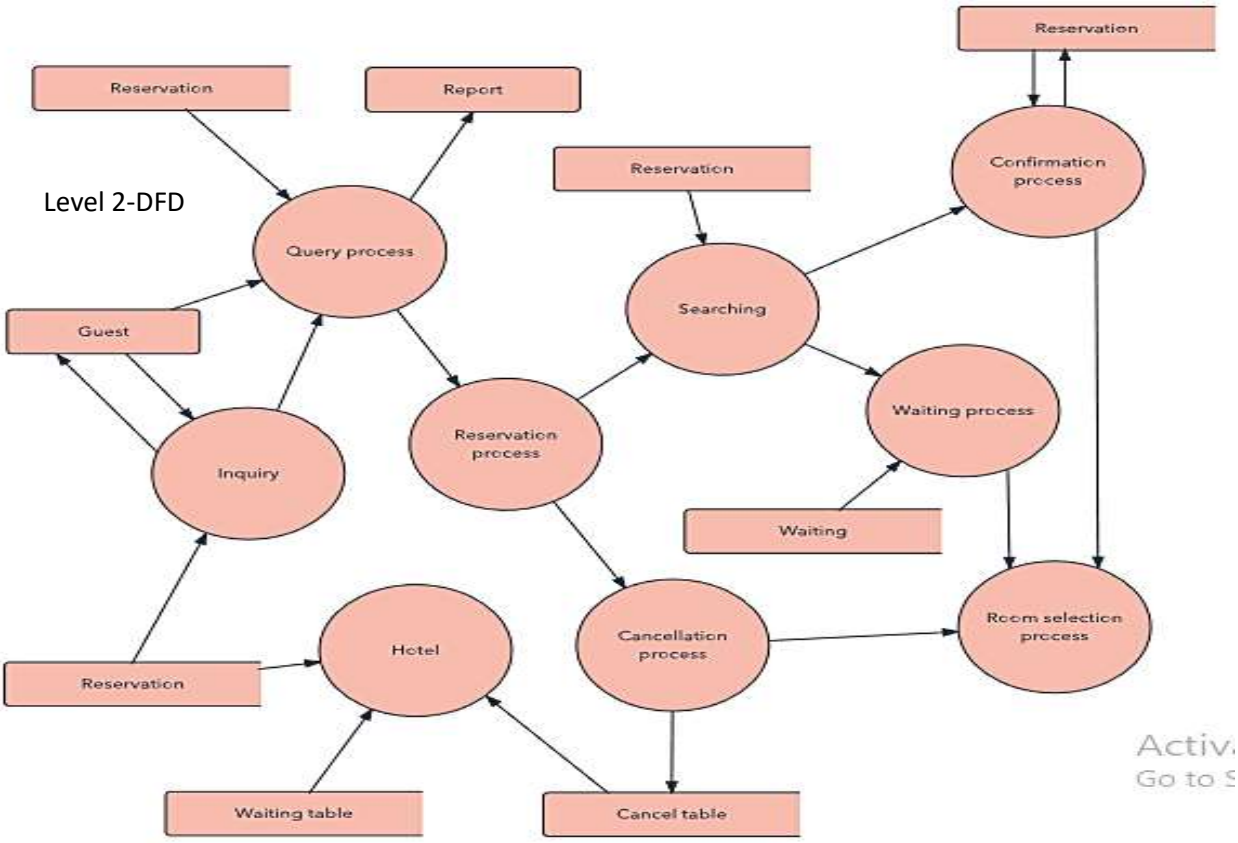


- **Level 1** - The Level 0 DFD is broken down into more specific, Level 1 DFD. Level 1 DFD depicts basic modules in the system and flow of data among various modules. Level 1 DFD also mentions basic processes and sources of information.
- **Level 2** - At this level, DFD shows how data flows inside the modules mentioned in Level 1. Higher level DFDs can be transformed into more specific lower level DFDs with deeper level of understanding unless the desired level of specification is achieved.

Level 1-DFD



Level 2-DFD



2. Data Dictionary

A data dictionary is a structured repository of data elements in the system. It stores the descriptions of all DFD data elements that is, details and definitions of data flows, data stores, data stored in data stores, and the processes.

قاموس البيانات هو مستودع منظم data elements في النظام .وهو يخزن الأوصاف لجميع DFD data elements، التي هي التفاصيل والتعاريف لتدفقات البيانات ومخازن البيانات والبيانات المخزنة في مخازن البيانات، والعمليات .

A data dictionary has many advantages:-

- **Documentation:** it is a valuable reference in any organization.
- التوثيق: انه مصدر ثمين في اي مؤسسة
- **Improves the communication between the analyst and the user**. by establishing consistent definitions of various elements, terms and procedures.
- تحسين التواصل بين المحلل والمستخدم: وذلك من خلال وضع تعاريف متسقة لمختلف العناصر والمصطلحات والاجراءات.
- **It plays an important role in building a database.** Most data base management systems (DBMSs) have a data dictionary as a standard feature.
- معظم قواعد البيانات لديها قاموس بيانات كميزة قياسية.

For example, refer the following table:

Sr. No.	Data Name	Description	No. of Characters
1	ISBN	ISBN Number	10
2	TITLE	Title	60
3	SUB	Book Subjects	80
4	ANAME	Author Name	15

There are three classes of items in data dictionary to be defined:

1. Data element: The smallest unit of data that provides for no further decomposition. For example, " date" consists of day, months and year. They hand together for all practical purposes. The description of a data element should include the name, description and an alias (synonym).

2. Data structure: a group of data elements handled as a unit. For example, "BOOK DETAILS" is a data structure consisting of the data elements author name, title, ISBN (International Standard Book Number), LOCN (Library of Congress Number), publisher's name and quantity. To describe any data structure by specifying the name of each data structure and the elements it represents, provided they are defined else- where in the data dictionary.

3. Data flows and data stores. As defined earlier, data flows are data structures in motion, whereas data stores are data structures at rest. A data store is a location where data structures are temporarily located.

The contents of a data flow may be described by the name (s) of the data structures(s) that passes along it. A data store is described by the data structures found in it and the data flows that feed it or are extracted from it.

3. Decision Trees

Decision trees are a method for defining complex relationships by describing decisions and avoiding the problems in communication. A decision tree is a diagram that shows alternative actions and conditions within horizontal tree framework. Thus, it depicts which conditions to consider first, second, and so on.

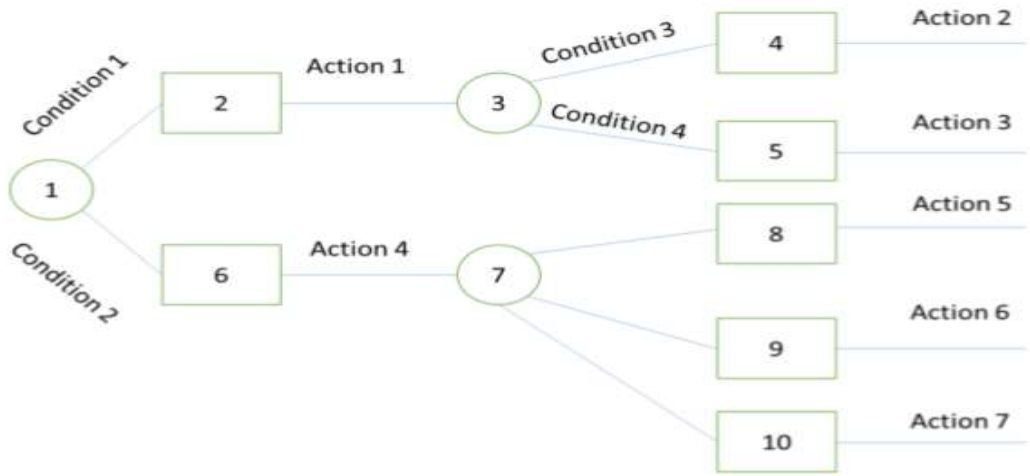
هي طريقة لتعريف العلاقات المعقدة من خلال وصف القرارات وتجنب المشاكل التواصل بين محلل النظم والمستخدم والمصمم . decision tree هي شكل يوضح الحلول والشروط البديلة ضمن إطار افقي بهيئة الشجرة . وهكذا، فإنه يصور أي شرط ممكن اخذه بنظر الاعتبار أولا، والثاني، وهكذا .

The root of the tree, on the left of the diagram, is the starting point of the decision sequence. The particular branch to be followed depends on the conditions that exist and the decision to be made. Progression from left to right along a particular branch is the result of making a series of decisions.

اساس (جذر) الشجرة، على يسار الشكل، هو نقطة البداية لتسلسل القرار .ويعتمد التفرع المعني الذي يتعين اتباعه على الشرط القائم والقرار الذي يتعين اتخاذه .التقدم من اليسار إلى اليمين على طول فرع معين هو نتيجة لاتخاذ سلسلة من القرارات.

Decision trees depict the relationship of each condition and their permissible actions. A square node indicates an action and a circle indicates a condition. It forces analysts to consider the sequence of decisions and identifies the actual decision that must be made.

Decision trees تصور العلاقة بين كل شرط وحلوله المسموح بها .تشير العقدة المربعة إلى حل معين والدائرة تشير إلى شرط .وهو يجبر المحللين على النظر في تسلسل القرارات وتحديد القرار الفعلي الذي يجب اتخاذه.



Decision Tree – Generalized Structure

Developing decision trees is beneficial to analysts in two ways.

- forces analysts to formally identify the actual decision that must be made.

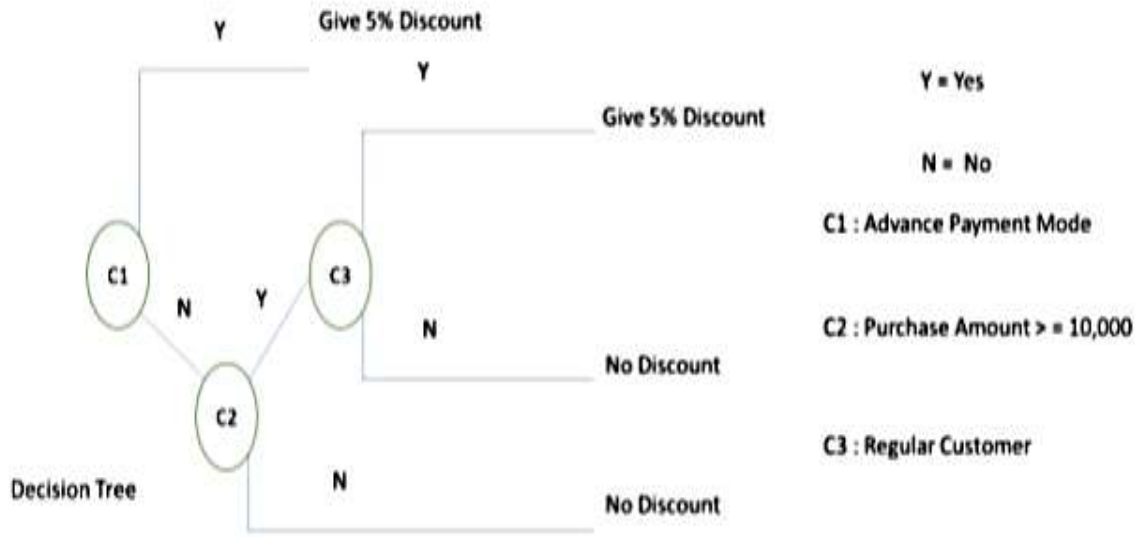
- force analysts to consider the sequence of decisions.

The major limitation of a decision tree is that it lacks information in its format to describe what other combinations of conditions you can take for testing. It is a single representation of the relationships between conditions and actions. For example, refer the following decision tree:

وتطوير decision trees مفيدة للمحللين بطريقتين.

- يجبر المحللين على تحديد القرار الفعلي الذي يجب اتخاذه.
- يجبر المحللين على الاخذ بنظر الاعتبار تسلسل القرارات.

المحدد الرئيسي decision tree هو أنها تفتقر إلى معلومات بهيئتها لوصف ما هي المجموعات ألاخرى من الشروط التي يمكنك اتخاذها للاختبار .وهي تمثيل واحد للعلاقات بين الشروط والحلول. على سبيل المثال، شجرة القرارات التالية:



4. Decision Tables

Decision tables are a method of describing the complex logical relationship in a precise manner which is easily understandable.

جداول القرار هي طريقة لوصف العلاقة المنطقية المعقدة بطريقة دقيقة يسهل فهمها.

A decision table is a table of contingencies for defining a problem and the actions to be taken. It is single representation of the relationships between conditions and actions.

جدول القرارات هو جدول للطوارئ لتحديد المشكلة والحلول المناسبة الواجب اتخاذها .وهو تمثيل واحد للعلاقات بين الشروط والحلول المناسبة.

- It is useful in situations where the resulting actions depend on the occurrence of one or several combinations of independent conditions.
- من المفيد في الحالات التي تعتمد فيها الحلول الناتجة على حدوث مجموعة واحدة أو عدة مجموعات من الشروط المستقلة.

- It is a matrix containing row or columns for defining a problem and the actions.
- وهي مصفوفة تحتوي على صفوف أو أعمدة لتعريف المشكلة والحلول المناسبة .

Components of a Decision Table

- **Condition Stub:** It is in the upper left quadrant which lists all the condition to be checked.
- هو في الربع الأيسر العلوي الذي يحدد كل الشروط ليتم التحقق منها.
- **Action stub:** It is in the lower left quadrant which outlines all the action to be carried out to meet such condition.
- هو في الربع السفلي الأيسر الذي يحدد كل الحلول التي يتعين القيام بها لتلبية هكذا شرط.
- **Condition Entry:** It is in upper right quadrant which provides answers to questions asked in condition stub quadrant.
- هو في الربع الأيمن العلوي الذي يوفر إجابات على الأسئلة التي طرحت في الربع الخاص ب Condition Stub .
- **Action Entry:** It is in lower right quadrant which indicates the appropriate action resulting from the answers to the conditions in the condition entry quadrant.
- هو في الربع الأيمن السفلي الذي يشير إلى الحل المناسب الناجم عن الإجابات على الشروط في الربع الخاص ب . condition entry

The entries in decision table are given by Decision Rules which define the relationships between combinations of conditions and courses of action. In rules section.

المدخلات في جدول القرارات تعطى بواسطة Decision Rules التي تعرف العلاقات بين مجموعات الشروط ومجموعات من الحلول .في قسم rules ،

- Y shows the existence of a condition. يظهر Y وجود شرط
- N represents the condition, which is not satisfied. يمثل N الشرط الذي هو غير متحقق
- A blank - against action states it is to be ignored. الفراغ بالمقارنة مع حالات الحل ينص عل ان يتم تجاهلها
- X (or a check mark will do) against action states it is to be carried out.

X(أو علامة الاختيار سوف تفعل) بالمقارنة مع حالات الحل ينص على أن يتم تنفيذها.

For example, refer the following table:

CONDITIONS	Rule 1	Rule 2	Rule 3	Rule 4
Advance payment made	Y	N	N	N
Purchase amount = Rs 10,000/-	-	Y	Y	N
Regular Customer	-	Y	N	-
ACTIONS				
Give 5% discount	X	X	-	-
Give no discount	-	-	X	X

5. Structured English

Structure English is derived from structured programming language which gives more understandable and precise description of process. It is based on procedural logic that uses construction and imperative sentences designed to perform operation for action.

اللغة الإنكليزية المهيكلة هي مشتقة من اللغات البرمجية المهيكلة التي تعطي وصفا أكثر تفهما ودقة للعملية .وهي مبنية على اساس المنطق الإجرائي الذي يستخدم اي من الاساليب التالية الجمل الامرية(MOVE,ADD) اوالجمل القرارية (IF THEN ELSE) اوالجمل التكرارية (Loop) المصممة لتأدية العمليات للحصول على الحل.

It is best used when sequences and loops in a program must be considered and the problem needs sequences of actions with decisions.

انها تستخدم بشكل جيد عندما السلاسل والحلقات التكرارية في البرنامج يجب الاخذ بها بنظر الاعتبار والمشكلة تحتاج سلسلة من الحلول مع القرارات.

It does not have strict syntax rule. It expresses all logic in terms of sequential decision structures and iterations.

ليس لديها قاعدة بناء جملة صارمة .وهو تعبرعن كل المنطق من حيث مصطلح هياكل القرار المتسلسلة والتكرار.

For example, see the following sequence of actions:

```
if customer pays advance
    then
           Give 5% Discount
    else
           if purchase amount >=10,000
                then
                      if  the customer is a regular customer
                            then Give 5% Discount
                             else  No Discount
                      end if
                else No Discount
          end if
 end if
```

6. Pseudocode

A pseudocode does not conform to any programming language and expresses logic in plain English.

pseudocode لا يتوافق مع أي لغة برمجة ويعبر عن المنطق باستخدام اللغة الإنجليزية الصريحة .

- It may specify the physical programming logic without actual coding during and after the physical design.
  - قد تشبه منطق البرمجة بشكل مباشر دون كتابة الكود الفعلي أثناء وبعد التصميم المادي.
- It is used in conjunction with structured programming.
  - يتم استخدامه جنبا إلى جنب مع البرمجة المهيكلة.
- It replaces the flowcharts of a program. انها تستبدل المخططات الانسيابية للبرنامج

Example: Program to print Fibonacci up to n numbers.

```
void function Fibonacci
Get value of n;
Set value of a to 1;
Set value of b to 1;
Initialize I to 0
for (i=0; i< n; i++)
{
   if a greater than b
   {
      Increase b by a;
      Print b;
   }
   else if b greater than a
   {
      increase a by b;
      print a;
   }
}
```

Guidelines for Selecting Appropriate Tools

Use the following guidelines for selecting the most appropriate tool that would suit your requirements:

- Use DFD at high or low level analysis for providing good system documentations.
- Use data dictionary to simplify the structure for meeting the data requirement of the system.
- Use structured English if there are many loops and actions are complex.
- Use decision tables when there are a large number of conditions to check and logic is complex.
- Use decision trees when sequencing of conditions is important and if there are few conditions to be tested.