**Chapter 4: System design**

|  |  |  |  |
| --- | --- | --- | --- |
| **4.1** | **Introduction about system design** | **…………………………………..** | **64** |
| **4.2** | **Entity relationship diagram (ERD)** | **…………………………………..** | **64** |
| **4.3** | **Class diagram** | **…………………………………..** | **67** |
| **4.4** | **Activity diagram** | **…………………………………..** | **70** |
| **4.5**  **4.6** | **Business model**  **Summery** | **…………………………………..**  **…………………………………..** | **73**  **74** |

**Chapter 4**

**System design**

**In this chapter you’ll get a detailed knowledge about the system design and database of the project.**

**4.1 Introduction about system design:**

The third phase of SDLC. Coming after planning and analyzing the system, it also includes as the most creative and hard phase while working on the system.

The design phase bridges the gap between problem domain and the existing system in a manageable way” [6]”, by showing the user interface, forms, and reports that will be used; and the specific programs, databases, and files that will be needed.

The two main kind of design is: logical design which represents the data flow, inputs, and outputs of the system. And physical design that concerned with user interface, process design, and data design.

**4.2 Entity Relationship Diagram ERD:**

An entity relationship diagram (ERD) shows the relationships of entity sets stored in a database. An entity in this context is an object, a component of data. An entity set is a collection of similar entities. These entities can have attributes that define its properties.

Entity is an object/concept about which you want to store information, represented by rectangle, attribute is characteristic of the entity, represented by oval shape:

* **Simple: having one value**

Simple attribute symbol

* **Multi value: having more than one value**

Multi value attribute symbol

* **Derived: based on another attribute**

Derived attribute symbol

**Relationship shows association between entities:**

**Cardinality:**

* **one to one relationship**

1

1

One to one relationship

* **one to many relationships**

1000000000000000000000

M

One to many relationships

* **many to many relationship**

M

M

Many to many relationships

**Participation:**

* **When A’s occurrence doesn’t depend on B’s occurrence, it called “PARTIAL”:**

A

B

Partial relationship

* **When A’s occurrence depends on B’s occurrence, it called “Mandatory”**

B

A

Mandatory relationship

**Diagram

Description automatically generated**

**Figure (4.1) entity relationships diagram (ERD)**

**4.3 Class Diagram:**

Class diagram is a structural UML diagram that show classes, relationships, interface, association, and collaboration.

Classes in the class diagram are the building blocks based on OOPs.

**4.3.1 Structure of class:**

* Class Name
* Attribute Structure shown in figure (4.2):
* modifier: represent visibility.

+: Public

-: Private

#: Protected

* Attribute-variable- name.
* attribute type (int, string …)
* Default value-optional-
* other property**.**

**Modifier attribute name: attribute type=”default value” {other property}**

Figure (4.2): Attribute Structure

* Operation Structure as shown in figure (4.3):
* Modifier.
* operation name.
* parameters.
* return value**.**

**Modifier operation name (parameters): return value**

Figure (4.3): Operation Structure

**Here is the whole class structure shown in figure (4.4):**

Table

Description automatically generated

Figure (4.4): Class Structure

**4.3.2 Relationships:**

* association: relation between instances of two classes.
* generalization/inheritance: relation between one general (super class) and specific (sub class) classes -kind-of relationship-
* aggregation: “Has-a”/ “whole/part” relation**.**

Whole part

Figure (4.5): aggregation relationship

* **composition: strong form of aggregation that implies ownership.**

Figure (4.6): Composition relationship

**This figure shows the class diagram of national health care system which contain 14 classes, each class has its own properties and relationships**

**Diagram

Description automatically generated**

**Figure (4.7) class diagram**

**4.4 Activity Diagram:**

**Activity diagram is a behavioral diagram used for business modeling and considered as an object oriented for flow chart and data flow, the basic notations used to draw activity diagram are:**

* **Initial/start state: shown in figure (4.8).**

Figure (4.8): Initial state symbol

* **Stop state: ending point, shown in figure (4.9).**

Figure (4.9): stop state symbol

* **Activity: task/process, shown in figure (4.10).**

Figure (4.10): Activity symbol

* **Decision: Boolean expression, shown in figure (4.11).**

Yes

No

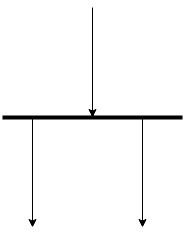
****

Figure (4.11): Decision symbol

* **Fork: one incoming transition and two or more outgoing transitions, shown in figure (4.12).**

Figure (4.12): Fork symbol

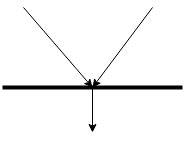
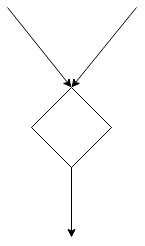
* **Join: several flow in, and one leaving, shown in figure (4.13).**

Figure (4.13): join symbol

****

* **Merge: shown in figure (4.14).**

Figure (4.14): Merge symbol

**The difference between merge and join is that the inputs must executed first to execute the output in join, but executing only one activity from input made the output executed in merge.**

* **Swimlane: vertical/horizontal zone represent class/dept., shown in figure (4.15).**

Swimlane name

Figure (4.15): Swimlane symbol

**The figure below shows the activity diagram that clarify the data flow and decisions taking through the system.**

**Diagram

Description automatically generated**

**Figure (4.16) activity diagram**

**4.5 Business Model:**

Business Model Canvas is one of the revolutionary tools that caters to the need for entrepreneurs to plan their projects and startups on paper before launching them, and it is a graphical visual method that can be used during the planning phase of the project in order to correct errors as much as possible.

The canvas has nine elements:

Customer Segments: the targeted group (people/organization), Customer relationship: self/Automated service, Channels: the way you’ll spread your idea with and communicate with customer segments, Value Propositions: the benefits coming from solving this problem, Key activities: actions/activities, Key resources: physical materials and skills used to get results, Key partners: people/organizations participating as

Partner with contracts maybe to help, Cost structure: your budget and the things you’ll pay for and revenue Stream: profits either material or morality.

Timeline

Description automatically generated with medium confidence

**Figure (4.17) business model**

**4.6 Summery:**

Finally, we include the structural and behavioral diagrams to more explanation of the system, which is ready now to real implementation, and that what we going to show in the next chapter.