**Chapter 4: System Planning**

Chapter 4 is representing the project planning about the system I have been developed. It is also describes about the Project Planning, project estimation, function point estimation, Task Scheduling, Project Scheduling Chart and cost estimation of the system.

**4.1 Functions of Proposed System**

1. Student Online admission [F01]
2. Login into the System [F02]
3. Student profile [F03]
4. Result management [F04]
5. Syllabus Management [F05]
6. Notice Management [F06]
7. Certificate Management [F07]
8. Research Publication [F08]
9. Published Book online sell [F09]
10. Report Generation [F10]

**4.1.1 Function Description**

Function description descriptive the function in details. It concerns on three factors: what is the possible input, possible output for a particular function and which table of the database uses by that function.

**Student Online Admission:**

Input: First Name, Last Name, Address, Image, Date of Birth, Email, Education Result of (SSC,HSC, Honours, Masters), Payment details.

Output: Student will get a password though email. Using this password, student can login the system and get a his/her profile, if admin confirmed his admission. Otherwise he get a waiting message. If admin cancel the admission request then he got an email with an explanation.

Use table of the database: tbl\_student

**Login into the System:**

Input: email, password

Output: Student get his profile and admin will get admin panel, if email and password match with database table data. Otherwise student and admin get error message.

Use table of the database: tbl\_admin, tbl\_student.

**Student Profile:**

Input: email, password.

Output: Student get student Profile, where student’s all information including results will be available.

Use table of the database: tbl\_student, tbl\_result.

**Result Management:**

Input: student ID, Subject code, and Marks.

Output: Every student can see his own result through his profile. Admin can see all result together of specific result.

Use table of the database: tbl\_result, tbl\_student, tbl\_subject.

**Syllabus Mangement:**

Input: Subject ID, Subject Name, Subject Description.

Output: Student can see his syllabus from his profile.

Use table of the database: tbl\_syllabus

**Notice Management:**

Input: Notice Text

Output: Notice will go to every student profile.

Use table of the database: tbl\_instruction.

**Certificate Management:**

Input: Student only click on Certificate option.

Output: Student will get his own certificate as PDF format.

Use table of the database: tbl\_certificate

**Research publication:**

Input: Research title, image(optional), Research text. Publication status(for admin).

Output: Research will be appear on website, If publication status is yes.

Use table of the database: tbl\_post

**Published Book Online Sell:**

Input: Click on book, Book Quantity, customer signing, customer login, shipping info, cart info.

Output: Get the buy confirmation message.

Use table of the database: tbl\_product,tbl\_customer,tbl\_shipping,tbl\_order, tbl\_order-details,tbl\_payment.

**Report Generate:**

Input: student ID, From Date, To Date, Subject code.

Output: Admin will get specific report

Use table of the database: tbl\_student, tbl\_result, tbl\_order, tbl\_customer.

**4.2 System Project Planning**

Software project planning is the second activity of CPF. Software project management commences with a set of activities that collectively called software project planning. Before starting any project, it is compulsory to estimate the work to be done, the resources that will be required, the time will elapse from start to finish and to analyze the project to determine whether it is feasible or not.

The following activities of software project planning that have followed in this project are:

* System Project Estimation
* Function Oriented Metrics
* Process Based Estimation
* Effort Distribution
* Task Scheduling
* Project Schedule Chart
* Cost Estimation

**4.2.1 System Project Estimation**

The accuracy of a software project estimate predicated based on a number of things:

1. Properly estimated the size of the product to build.
2. The ability to translate the size estimation into human effort, calendar time and money.
3. The degree to which the project plan reflects the abilities of the software team or engineer.
4. The stability of the product requirements and the environment that supports the software engineering effort.

Software size estimation is the most important matter that I have to consider during the software project. If the software size not calculate properly, then this will cause various problems such as scheduling problems, budget problem etc. As the project goes on. Before estimating the software size, I have to confirm that software scope is bounded.

**4.2.2 Function Oriented Metrics**

Function point based estimation focuses on information domain values rather that software values. Function points are computed by comparing five information domain characteristics. The information domain values are as follows

**Number of external inputs** – Each user input that provides distinct application-oriented data to the software is counted inputs should be distinguished from inquires.

**Number of external outputs** – Each user output that provides application-oriented information to the user is counted.

**Number of external inquires** – An inquiry defined as an on-line input those results in the generation of some immediate software response in the form of an on-line output. Each distinct inquiry counted.

**Number of files –** Each logical master file counted.

**Numbers of external interfaces** – All machine-readable interfaces that used to transmit information to another system counted.

The weights of the domains are fixes, which are provided in appropriate table location. Weights can be divided into three categories according to the functionality of the system. They are simple, average and complex. The total system is a complex system but the part of the total system. Once these data has collected, a complexity value is associated with each count.

To find out the FP count the following formula is used,

FP Count = (((4 \* Most Likely) + Optimistic + Pessimistic) / 6) \* Weight)

To compute function points (FP), the following relationship is used:

FP = Count Total \* [0.65 + 0.01 \* Σ (Fi)]

Complexity adjustment factor = [0.65+0.01\*Sum of factor values]

FP estimated =count total\*Complexity adjustment factor.

Function Point Estimation = Total FP estimated/No. of function point.

The count total is the sum of all FP entries.

* **FP count:**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Domain** | **Optimistic** | **Likely** | **Pessimistic** | **Est. Count** | **Weight** | **FP Count** |
| Number of External Input | 9 | 4 | 11 | 6.00 | 3 | 18.00 |
| Number of External Output | 20 | 8 | 20 | 12.00 | 4 | 48.00 |
| Number of External Inquiries | 6 | 3 | 6 | 4.00 | 3 | 12.00 |
| Number of Internal Logical Files | 24 | 13 | 20 | 16.00 | 7 | 112.00 |
| Number of External Interface | 15 | 7 | 11 | 9.00 | 5 | 45.00 |
| **Count Total:** | | | | | | 235.00 |

Figure 4.1: FP Count

* **Complexity adjustment value:**

|  |  |  |
| --- | --- | --- |
| Number | Factor | Value |
| 1 | Does the system require reliable backup and recovery? | 2 |
| 2 | Are specialized data communications required? | 3 |
| 3 | Are there any distributed processing functions? | 0 |
| 4 | Is performance critical? | 1 |
| 5 | Does the system run in existing operational environment? | 0 |
| 6 | Does the system require off-line data entry? | 3 |
| 7 | Input transaction over multiple screens | 0 |
| 8 | Are the master files updated on-line? | 2 |
| 9 | Are the input, output, files or inquiries complex? | 0 |
| 10 | Is the internal processing complex? | 0 |
| 11 | Is the code designed to be reusable? | 4 |
| 12 | Are conversations or installation included in the design? | 0 |
| 13 | Is multiple designed for change? | 0 |
| 14 | Is the system designed to facilitate change and case? | 4 |
|  | Σ (Fi) | 19 |
| Complexity Adjustment Factor = 0.84 | | | |

|  |
| --- |
| FP Estimated = 197.4 |

Figure 4.2: Complexity Adjustment Table

**4.2.3 Function Point Estimation**

Here FP Estimated = 197.4,where the FP is estimated using the formula:

FP Estimated = Count Total \* Complexity Adjustment factor.

Function Point Estimation = Total FP Estimated / No. of function point.

=197.4 / 10 =**19.74 man month.**

**= 20 man month (Approximate).**

Time Frame Calculation = 20 / 05

**= 4 Months**

**4.2.4 Process Based Estimation**

In process-based estimation, process is decomposed into a relatively small set of tasks and the effort required to accomplish each task is estimated. Process based estimation begins with a delineation of software functions obtained from the project scope. A series of software process activities must be performed for each function.

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Activity** | **CC** | **Planning** | **Engineering** | | **Construction** | | **Imp.** | **Total** |
| **Function** |  |  | **Analysis** | **Design** | **Code** | **Test** |  |  |
| **F1** |  |  | 0.12 | 0.15 | 1.25 |  |  |  |
| **F2** |  |  | 0.17 | 0.12 | 1.15 |  |  |  |
| **F3** |  |  | 0.19 | 0.25 | 0.45 |  |  |  |
| **F4** |  |  | 0.12 | 0.25 | 0.45 |  |  |  |
| **F5** |  |  | 0.60 | 0.25 | 0.75 |  |  |  |
| **F6** |  |  | 0.54 | 0.25 | 0.75 |  |  |  |
| **F7** |  |  | 0.36 | 0.25 | 0.45 |  |  |  |
| **F8** |  |  | 0.12 | 0.25 | 0.45 |  |  |  |
| **F9** |  |  | 0.54 | 0.55 | 1.15 |  |  |  |
| **F10** |  |  | 0.24 | 1.4 | 2.00 |  |  |  |
| **Total** | **1.00** | **1.50** | **3.00** | **3.47** | **8.85** | **1.55** | **0.55** | **19.92** |
| **Effort** | **2%** | **6%** | **19%** | **27%** | **36%** | **6%** | **4%** | **100%** |

Figure 4.3: Process Based Estimation

**Figure 4.4 :** Effort Based Estimation

**4.2.5 Time Line Calculation**

Process Based Estimation = **19.92**man months

Estimated time for the project = Estimated Man Month / No. of People Working.

= (19.92 / 5) months

**= 3.984 ≈ 4 months need for 5 people to complete this system.**

**4.2.6 Effort Distribution**

The project estimation technique leads to estimates of work units required to complete the software development. A recommended distribution of effort across the definition and development phases referred as the 40-20-40 rule. Forty percent of all effort allocated to front-end analysis and design, twenty percent allocated to coding and the remaining forty percent allocated to back-end testing. This rule used as a guideline only.

In this project, 46% of full software development has been allocated to analysis and design, 36% has allocated to coding and the remaining 18% is allocated to software testing and support.

**Figure 4.4 :** Effort Based Estimation

**Description:**

* 1 (6% - Planning)
* 2 (2% - Customer Communication)
* 3 (27% - Analyzing)
* 4 (19% - Designing)
* 5 (36% - Coding)
* 6 (6% - Testing).
* 7 (4% - Implementation).

**4.2.7 Task Scheduling**

Project scheduling is an activity of distributing the estimated efforts within the planned project duration. There are some basic rules for project scheduling. They are as follows –

**Compartmentalization** – The project must compartmentalize into a number of manageable activities and tasks.

**Interdependency** – The interdependency of each compartmentalized activity or task must be determined. Some tasks must occur in sequence while others can occur in parallel.

**Time allocation** – Each task to be scheduled must allocated some number of work units.

**Effort validation** – Every project has a defined number of staff members. It should ensure that no more than the allocated number of people has scheduled at any given time.

**Defined responsibilities** – Every task that is scheduled should assign to a specific team member.

**Defined outcomes** – Every task that is scheduled should have a defined outcome. The outcome is normally a work product or a part of a work product.

**4.2.8Project Schedule Chart**

Total system development is a combination of set of tasks. These set of tasks should done sequentially and timely. Project schedule works as the guideline of the system developer. The following is the schedule chart of this project:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Category** | **1st Month** | **2nd Month** | **3rd Month** | **4th Month** |
| **Customer Communicator** |  |  |  |  |
| **Planning** |  |  |  |  |
| **Analysis** |  |  |  |  |
| **Design** |  |  |  |  |
| **Coding** |  |  |  |  |
| **Testing** |  |  |  |  |
| **Implementation** |  |  |  |  |

Figure4.5: Project Schedule Chart

**4.2.9 Cost Estimation**

* Personnel Cost
* Software Cost
* Hardware Cost

We calculate cost by double decline method. How we calculate the depreciation is given on the bellow chart.

**Double Decline Method**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Item Name** | **Year** | **Value** | **Depreciation** | **Depreciated Value** |
| Computer | 1st | 50000 | 25000 | 25000 |
| 2nd | 25000 | 12500 | 12500 |
| 3rd | 12500 | 6250 | 6250 |
| 4th | 6250 | 3125 | 3125 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Item Name** | **Year** | **Value** | **Depreciation** | **Depreciated Value** |
| Windows 7 | 1st | 20000 | 10000 | 10000 |
| 2nd | 10000 | 5000 | 5000 |
| 3rd | 5000 | 2500 | 2500 |
| 4th | 2500 | 1250 | 1250 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Item Name** | **Year** | **Value** | **Depreciation** | **Depreciated Value** |
| Adobe Dream Weaver CS4 | 1st | 10000 | 6700 | 3300 |
| 2nd | 3300 | 1320 | 1980 |
| 3rd | 1980 | 792 | 1188 |

**Personnel Cost:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Type** | **Number** | **Month** | **Salary Per Month** | **Total** |
| CC | 1 | 0.5 | 10000 | 5000 |
| System Analyst | 1 | 3.5 | 50000 | 175000 |
| Senior Developer | 1 | 1.5 | 30000 | 45000 |
| Junior Developer | 1 | 1 | 20000 | 20000 |
| Tester | 1 | 1.5 | 15000 | 22500 |
| **Total:** | | | | 267500 |

**Software Cost:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Item#** | **Year** | **Value** | **Depreciation** | **Depreciated Value** |
| Windows 7 | 2nd | 10000 | 5000 | 5000 |
| Adobe DW CS4 | 2nd | 3300 | 1320 | 1980 |
| **Total:** | | | | 6980 |

**Hardware Cost:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Item#** | **Year** | **Value** | **Depreciation** | **Depreciated Value** |
| Computer | 2nd | 25000 | 12500 | 12500 |
| **Total:** | | | | 12500 |

**Total System Development Cost:**

|  |  |
| --- | --- |
| **Cost Type** | **Cost** |
| **Personnel Cost** | **2, 67,500 BDT.** |
| **Software Cost** | **6,980 BDT.** |
| **Hardware Cost** | **12,500 BDT.** |
| **Total:** | **2, 86,980 BDT.** |