Project Management

PMG 4101

Section: B

PawPal

Team SteelFrags

Md. Rakibur Rahman Lipu - 011221142
Abdullah Abrar Labib - 011221150
Khan Israk Ahmed - 011221275
Zawed Bishwas - 011221565
Maharab Hossain - 011221608

Group : 07

Submitted to
Kazi Abdun Noor
Lecturer, Dept. of CSE

Index

1. Abstract

PawPal is a unique, all-in-one digital ecosystem that simplifies and improves pet care for today's busy, urban pet owners. Recognizing that pets are valued family members who deserve the best, PawPal bridges the gap between fragmented pet services by combining veterinary care, grooming, pet walking, pet product marketplaces, and an engaged community into a single, user-friendly platform.

PawPal enables owners to make informed decisions about their pet's health and well-being by utilizing modern technologies, such as tele-veterinary consultations, AI-powered symptom checkers, individualized pharmaceutical recommendations, and geolocation-based emergency support. The platform was thoughtfully designed for the regional context of Bangladesh, incorporating popular local payment alternatives such as bKash and Nagad, as well as supporting the Bengali language to provide accessibility for all.

Beyond on-demand care and commerce, PawPal fosters a supportive pet-loving community in which owners may share their knowledge and experiences. Its unified, trusted environment strives to inspire ethical pet ownership, promote local pet businesses, and, eventually, make pet care more smooth, empathetic, and connected—ensuring that no pet goes uncared for and no pet owner feels abandoned.

2. Introduction

In today's fast-paced society, pets are more than simply animals; they are companions, family members, and emotional pillars. As urban lifestyles become more complicated, pet owners frequently find it difficult to provide continuous healthcare, grooming, and care for their furry (or feathered) animal companions. The traditional pet care ecosystem lacks a cohesive digital platform that addresses pet owners' different demands in an efficient, accessible, and user-friendly way.

By offering a comprehensive web-based system intended exclusively for pets and their owners, PawPal aims to close this gap. It combines veterinarian care, grooming assistance, purchasing medicines and accessories, emergency support, and an active pet owner community into a single, integrated digital platform.

PawPal offers services including online veterinarian consultations, location-based emergency hospital routing, pet nanny and pet walker hiring, AI-powered symptom analysis, smart prescription suggestions, and monthly health reminders, all of which are inspired by successful health-tech and service aggregation models. In order to provide safe and easy transactions, the platform is further localized with integrated support for well-known digital payment methods in Bangladesh, such as bKash and Nagad.

Whether a pet parent requires a quick vet visit, a new toy, or assistance while on vacation, PawPal provides a dependable, inventive, and heartfelt solution, guaranteeing that no pet goes uncared for and no pet owner feels abandoned.

3. Vision & Goal

To develop the most trusted and comprehensive digital ecosystem for pet care in Bangladesh and beyond, we will seamlessly integrate technology, community, and compassion into all aspects of a pet's life.

1. **Unify Pet Services Online:** Provide a single platform from which clients can access all pet-related services—medical, grooming, lifestyle, and community—without having to download various applications or visit multiple websites.
2. **Enhance Pet Healthcare:** Utilize features such as tele-veterinary consultation, disease-specific vet advice, and emergency locators to provide rapid and educated medical care.
3. **Encourage Responsible Pet Ownership:** Offer features such as monthly checkup reminders, individualized prescription recommendations, and access to nurturing services to improve day-to-day pet care.
4. **Encourage Pet-Related Businesses:** Establish a marketplace where nannies, walkers, veterinary professionals, and vendors of pet supplies may advertise their services and earn a steady income.
5. **Encourage a Pet-Loving Community:** Create a community forum that allows pet owners to ask questions, share experiences, and offer support to each other.
6. **Localize for Accessibility:** Make sure all features, such as offline prescription downloads, bKash/Nagad payments, and Bengali language support, are compatible with regional infrastructure and services.

4. Planning

Careful planning is required to ensure the PawPal platform is installed effectively and on time. A modular, milestone-driven methodology will be employed because this system is made up of multiple interconnected services. The project will be broken down into distinct phases to ensure that each crucial component is developed, tested, and implemented progressively.

Development Methodology

The Agile Software Development Lifecycle (SDLC) will be followed for this project, promoting iterative progress, flexibility, and quick adaptation to client or user feedback.

Each sprint will last approximately 2 weeks, with clear deliverables at the end of each cycle. To ensure performance and transparency, each sprint will include code reviews, quality assurance, and stakeholder demos.

4.1. Phases of the Project

Phase 1: Requirement Analysis

- Conduct interviews with pet owners, veterinary doctors, and pet service providers.
- Identify functional and non-functional requirements.
- Finalize feature set and technical specifications.

Phase 2: System Design

- Create system architecture diagrams.
- Design database schema and entity-relationship diagrams.
- Build UI/UX wireframes and navigation prototypes.
- Finalize API structure and third-party service plans (payment, maps, video calls).

Phase 3: Core Development

- **Week 1-2:** User Authentication, Profile Management, and Pet Profile Module.

  Week 3-4: Medicine Store (MediPaws) and Accessories Store (PawCessories) backend and frontend.
- **Week 5-6:** Vet Appointment Booking (VetConnect) and Disease-Specific Vet Filtering (DiagnoPaws).
- **Week 7:** Video Consultation Service Integration (PawMeet Live).
- **Week 8:** Payment System Integration (PayPaws) using bKash and Nagad APIs.

Phase 4: Advanced Features & Community Modules

- PetWalk Hero and PawSitter agent hiring module.
- Emergency Map-based Vet Locator (PawAlert).
- Community Forum (PawTalk Forum).
- Reminder System (VetPulse) for checkups and vaccinations.
- AI-powered Medicine Recommendation and Disease Detection (Beta).

Phase 5: Testing & Q/A

- Unit Testing for all modules.

- Functional Testing for cross-feature interactions.
- Usability Testing with sample users.
- Security & Payment Gateway Validation.

Phase 6: Deployment

- Deploy to staging server for pre-launch.
- Final bug fixes and performance optimizations.
- Deploy to production environment.

Phase 7: Maintenance & Future Enhancements (Ongoing)

- Continuous monitoring of server health and user feedback.
- Regular updates based on user demand and system logs.
- Planning of future enhancements like wearable device integration and mobile apps.

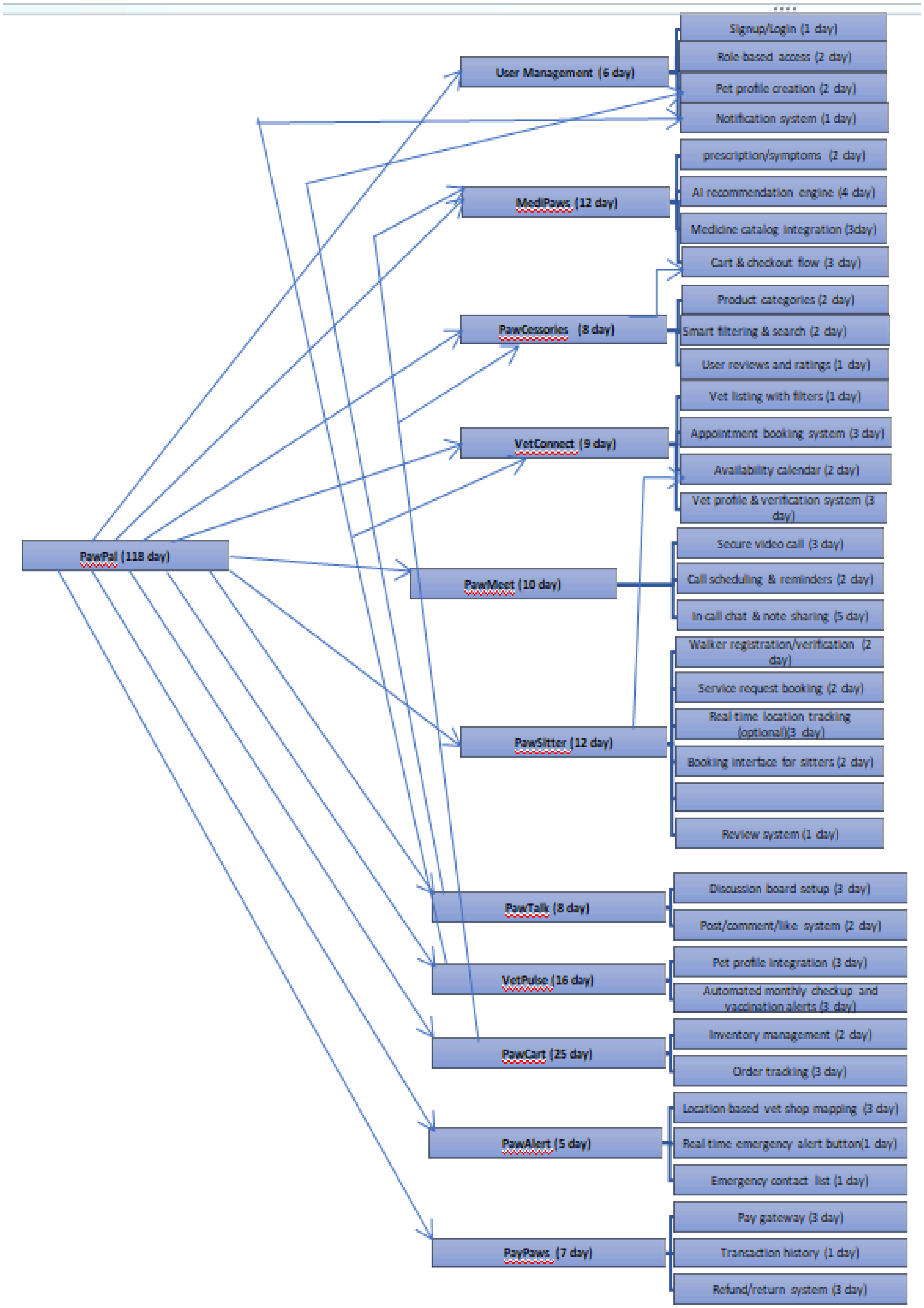
4.2. Resource Allocation

- Product Owner: 1
- Project Manager: 1
- Backend Developers: 2
- Frontend Developers: 2
- UI/UX Designer: 1
- QA Engineer: 1
- DevOps Engineer: 1
- Marketing & Community Lead: 1

4.3. Risk Mitigation

- Payment Security: Full encryption and sandbox testing for all transactions.
- Scalability: Cloud-native architecture to allow load balancing.
  Data Privacy: Implementation of secure login protocols and encrypted databases.
- Service Availability: Use of monitoring tools to detect and resolve downtime issues rapidly.

5. WBS Chart


PawPal (118 day)
User Management (6 day)
Signup/Login (1 day)
Role based access (2 day)
Pet profile creation (2 day)
Notification system (1 day)
MediPaws (12 day)
prescription/symptoms (2 day)
AI recommendation engine (4 day)
Medicine catalog integration (3day)
Cart & checkout flow (3 day)
PawCessories (8 day)
Product categories (2 day)
Smart filtering & search (2 day)
User reviews and ratings (1 day)
VetConnect (9 day)
Vet listing with filters (1 day)
Appointment booking system (3 day)
Availability calendar (2 day)
Vet profile & verification system (3 day)
PawMeet (10 day)
Secure video call (3 day)
Call scheduling & reminders (2 day)
In call chat & note sharing (5 day)
PawSitter (12 day)
Walker registration/verification (2 day)
Service request booking (2 day)
Real time location tracking (optional)(3 day)
Booking interface for sitters (2 day)
Review system (1 day)
PawTalk (8 day)
Discussion board setup (3 day)
Post/comment/like system (2 day)
VetPulse (16 day)
Pet profile integration (3 day)
Automated monthly checkup and vaccination alerts (3 day)
PawCart (25 day)
Inventory management (2 day)
Order tracking (3 day)
PawAlert (5 day)
Location based vet shop mapping (3 day)
Real time emergency alert button(1 day)
Emergency contact list (1 day)
PayPaws (7 day)
Pay gateway (3 day)
Transaction history (1 day)
Refund/return system (3 day)

6. Wideband Delphi Estimation Form for PawPal

Assigned Team Members:
Md. Rakibur Rahman Lipu,
Abdullah Abrar Labib,
Khan Israk Ahmed,
Zawed Bishwas,
Maharab Hossain

Date: 15.04.2025

Unit: Days

Task ID	Task Name	Assigned To	Duration	Vote
1	User Authentication & Profile Management	All team members	6 days	4
1.1	Signup/Login	Zawed	1 day	3
1.2	Role-based access (User, Vet, Admin)	Labib	2 days	2
1.3	Pet profile creation (species, age, health info)	Israk	2 days	3
1.4	Notification system	Maharab	1 day	3
2	MediPaws (AI e-Pharmacy)	Zawed,Labib,Maharab, Lipu	12 days	4

2.1	Upload prescription/symptoms	Zawed	2 days	3
2.2	AI recommendation engine	Labib	4 days	2
2.3	Medicine catalog integration	Maharab	3 days	3
2.4	Cart & checkout flow	Lipu	3 days	3
3	PawCessories (Accessory Store)	Zawed,Labib,Maharab, Israk	8 days	4
3.1	Product categories (clothes, toys, beds, etc.)	Israk	2 days	3
3.2	Smart filtering & search	Maharab	2 days	3
3.3	User reviews and ratings	Zawed	1 days	2
3.4	Cart & checkout flow	Labib	3 days	3
4	VetConnect (Vet Booking)	Zawed,Labib,Maharab,Lipu	9 days	4
4.1	Vet listing with filters	Lipu	1 days	3
4.2	Appointment booking system	Zawed	3 days	2

4.3	Availability calendar	Labib	2 days	2
4.4	Vet profile & verification system	Maharab	3 days	2
5	PawMeet Live (Video Consultation)	Labib,Maharab,Israk	10 days	4
5.1	Secure video call	Labib	3 days	3
5.2	Call scheduling & reminders	Maharab	2 days	2
5.3	In-call chat & note sharing	Israk	5 days	3
6	PawSitter (Pet Sitting)	Zawed,Labib,Maharab,Israk, Lipu	12 days	4
6.1	Walker registration/verification	Zawed	2 days	3
6.2	Service request booking	Labib	2 days	2
6.3	Real-time location tracking (optional)	Maharab	3 days	2
6.4	Booking interface for sitters	Israk	2 days	3
6.5	Availability calendar	Lipu	2 days	3

6.6	Review system	Labib	1 days	3
7	PawTalk Forum (Community Space)	Zawed,Labib,Maharab	8 days	4
7.1	Discussion board setup	Maharab	3 days	3
7.2	Post/comment/like system	Labib	2 days	2
7.3	Admin moderation tools	Zawed	2 days	2
8	VetPulse (Health Reminders)	Zawed,Labib,Maharab,Lipu	16 days	4
8.1	Pet profile integration	Lipu	3 days	3
8.2	Automated monthly checkup and vaccination alerts	Zawed	3 days	3
8.3	Notification settings	Labib	1 days	3
8.4	*4.VetConnect (Vet Booking)	Zawed,Labib,Maharab,Israk	9 days	3
9	PawCart+(Smart Store)	Zawed,Lipu,Maharab,Israk	25 days	4
9.1	Inventory management	Zawed	2 days	3

9.2	Order tracking	Lipu	3 days	2
9.3	*2.MediPaws (AI e-Pharmacy)	Zawed,Labib,Maharab,Israk	12 days	2
9.4	*3.PawCessories (Accessory Store)	Zawed,Labib,Maharab,Lipu	8 days	2
10	PawAlert(Emergency Map)	Zawed,Labib,Maharab	5 days	4
10.1	Location-based vet shop mapping	Zawed	3 days	3
10.2	Real-time emergency alert button	Labib	1 days	3
10.3	Emergency contact list	Maharab	1 days	2
11	PayPaws(Payment Gateway)	Zawed,Labib,Lipu	7 days	4
11.1	Pay gateway (bKash & Nagad integration)	Lipu	3 days	2
11.2	Transaction history	Zawed	1 days	2
11.3	Refund/return system	Labib	3 days	3

7. Function Point Analysis of our project

7.1. Function Point

A function point (FP) is a unit of measurement that quantifies the amount of functionality an application provides to the user. It's a way to measure the size of a software system based on the functionality it delivers. Function Point relies on 5 things:

1. **External inputs (EI)**
2. **External outputs (EO)**
3. **External inquiries (EQ)**
4. **External Interface Files (EIF)**
5. **Internal Logical Files (ILF)**

The size is determined by counting the number of these factors in the system and adjusting that total for the functional complexity of the system.

FP = UFC * TCF

7.2. Function Point Analysis(FPA)

Functional Point Analysis (FPA) is a software measurement technique used to assess the size and complexity of a software system based on its functionality. It involves categorizing the functions of the software, such as input screens, output reports, inquiries, files, and interfaces, and assigning weights to each based on their complexity. By quantifying these functions and their associated weights, FPA provides an objective measure of the software's size and complexity.

7.3. Types of Function Point Analysis

1. Transactional Functional Type

1. **External Input (EI):** EI processes data or control information that comes from outside the application's boundary. The EI is an elementary process.
2. **External Output (EO):** EO is an elementary process that generates data or control information sent outside the application's boundary.
3. **External Inquiries (EQ):** EQ is an elementary process made up of an input-output combination that results in data retrieval.

2. Data Functional Type:

1. **Internal Logical File (ILF):** A user-identifiable group of logically related data or control information maintained within the boundary of the application.
2. **External Interface File (EIF):** A group of users recognizable logically related data allusion to the software but maintained within the boundary of another software.

7.4. Advantages and Disadvantages of Function Point Analysis

7.4.1. Advantages of Functional Point Analysis

1. **Technological Independence:** It calculates a software system's functional size independent of the underlying technology or programming language used to implement it. As a result, it is a technology-neutral metric that makes it easier to compare projects created with various technologies.
2. **Better Accurate Project Estimation:** It helps to improve project estimation accuracy by measuring user interactions and functional needs. Project managers can improve planning and budgeting by using the results of the FPA to estimate the time, effort and resources required for development.
3. **Improved Interaction:** It provides a common language for business analysts, developers, and project managers to communicate with one another and with other stakeholders. By communicating the size and complexity of software in a way that both technical and non-technical audiences can easily understand this helps close the communication gap.
4. **Making Well-Informed Decisions:** FPA assists in making well-informed decisions at every stage of the software development life cycle. Based on the functional requirements, organizations can use the results of the FPA to make decisions about resource allocation, project prioritization, and technology selection.
5. **Early Recognition of Changes in Scope:** Early detection of changes in project scope is made easier with the help of FPA. Better scope change management is made possible by the 22 measurement of functional requirements, which makes it possible to evaluate additions or changes for their effect on the project's overall size.

7.4.2. Disadvantages of Functional Point Analysis

1. **Subjective Judgement:** One of the main disadvantages of Functional Point Analysis is its dependency on subjective judgement i.e. relying on personal opinions and interpretations instead of just using clear, measurable standards.
2. **Low Accuracy:** It has low evaluation accuracy as it's dependency on subjective judgement.
3. **Time Consuming:** Functional Point Analysis is a time consuming process, particularly during the initial stages of implementation.
4. **Steep Learning Curve:** Learning FPA can be challenging due to its complexity and the length of time required to gain proficiency.
5. **Less Research Data:** Compared to LOC-based metrics, there is relatively less research data available on function points.
6. **Costly:** The need for thorough analysis and evaluation can result in increased project timelines and associated costs.

7.5. The total calculation of FP in your project

This section calculates the Function points for the PawPal project.

The formula for calculating function point is *FP = UFC × TCF*; where *FP* is the Function Point, *UFC* is the Unadjusted Function Point Count and *TCF* is the Technical Complexity Factor. UFC is also known as UFP, Unadjusted Function Point and TCF is also known as VAF, Value Adjustment Factor or CAF, Complexity Adjustment Factor. To calculate Function Point for PawPal, we need to calculate UFC and TCF.

7.5.1. Unadjusted FP Count (UFPC)

The unadjusted function point UFP count in COCOMO is the initial calculation of the size of a software project based on its functionality. It's determined by counting the five types of functional components seen in the system and assigning a weight to each. The table shows all the five function counts for PawPal, considering all the features:

Feature	Ei (External Inputs)	EO (External Outputs)	EQ (External Queries)	ILF (Internal Logical Files)	EIF (External Interface Files)
Signup/Login	1			1	
Role-based access (User, Vet, Admin)	1			1	
Pet profile creation (species, age, health info)	1	1		1	
Notification system	1	1			

Upload prescription/symptoms	1			1	
AI recommendation engine		1			1
Medicine catalog integration		1	1		
Cart & checkout flow		1		1	
Product categories (clothes, toys, beds, etc.)		1	1		
Smart filtering & search			1		
User reviews and ratings	1				
Vet listing with filters			1		

Appointment booking system	1				
Availability calendar			1		
Vet profile & verification system		1	1	1	
Secure video call				1	1
Call scheduling & reminders		1			
In-call chat & note sharing				1	
Walker registration/verification	1			1	
Service request booking	1				
Real-time location		1			1

tracking (optional)					
Booking interface for sitters	1		1		
Review system	1				
Discussion board setup		1	1		
Post/comment/like system	1			1	
Admin moderation tools	1	1		1	
Pet profile integration	1	1		1	
Automated monthly checkup and vaccination alerts		1			
Notification settings	1				

Inventory management	1				
Order tracking		1			
Location-based vet shop mapping			1		
Real-time emergency alert button	1				
Emergency contact list	1	1	1	1	1
Pay gateway (bKash & Nagad integration)	1				1
Transaction history		1	1	1	
Refund/return system	1		1		

Metric	Total Count
Ei (External Inputs)	20
EO (External Outputs)	16
EQ (External Queries)	12
ILF (Internal Logical Files)	14
EIF (External Interface Files)	5

The equation of calculating unadjusted function point count UFC is:

$$\mathbf{UFP} = w\ \mathbf{EI} \times N\ \mathbf{EI} + w\ \mathbf{EO} \times N\ \mathbf{EO} + w\ \mathbf{EQ} \times N\ \mathbf{EQ} + w\ \mathbf{ILF} \times N\ \mathbf{ILF} + w\ \mathbf{EIF} \times N\ \mathbf{EIF}$$

Here, all the component counts N are determined. The following table is used to determine the weights.

Item	Simple	Average	Complex
EI	3	4	6
EO	4	5	7
EQ	3	4	6
ILF	7	10	15
EIF	5	7	10

In PawPal the EI is average,EO is average,EQ is average,ILF is average & EIF aslo average

UFP = 20 × 4 + 16 × 5 + 12 × 4 + 14 × 10 + 5 × 7 = 383

So, the Unadjusted Function Point Count is: 383

7.5.2. Technical Complexity Factor (TCF)

In COCOMO, the technical complexity factor TCF is a concept present in the Intermediate COCOMO model, not the Basic COCOMO which uses unadjusted function points UFP. The TCF refines the UFP count by considering additional factors that influence the inherent complexity of the software being developed. These factors go beyond the basic functionality and delve into technical characteristics that can make the project more challenging to build.

There are a total of 14 factors, each ranging from zero to five. The values represent the level of priority or importance in a system. The level of importance are given below:

- **0 = No influence**
- **1 = Incidental**
- **2 = Moderate**
- **3 = Average**
- **4 = Significant**
- **5 = Essential**

Each factor is assigned one of the points mentioned above. Below is the table that shows all the complexity factors assigned for PawPal :

Technical Factors	**Factor points**
F1 Reliable backup and recovery	4
F2 Data communications	5
F3 Distributed functions	2
F4 Performance	3
F5 Heavily used configuration	4
F6 Online data entry	3
F7 Operational ease	4
F8 Online update	3
F9 Complex interface	3
F10 Complex processing	4
F11 Reusability	3
F12 Installation ease	4
F13 Multiple sites	4
F14 Facilitate	5
∑F	51

In the table, ∑F shows the total summation of all factor points. The equation for calculating the technical complexity factor TCF:

$$\mathbf{TCF = 0.65 + (0.01 \times \sum F)}$$

As we calculated ∑F to be 51 , we can calculate the TCF by the equation. So,

TCF = 0. 65 + (0. 01 × 51) = 1.16

From this section, the Technical Complexity Factor is 1.16

7.5.3. Counting Function Point (FP)

To calculate function point FP, unadjusted function point count and technical complexity factor is multiplied.

- Unadjusted Function Point 383
- Technical Complexity Factor 1.16

So, Function Point(FP) is 383 × 1. 16 = 444.28

The Function pointer is 444.28

8. COCOMO Estimation for PawPal

For every project, COCOMO (COnstructive COst MOdel) is estimated to calculate the effort, time, and cost. This estimation helps in planning the project, resource allocation, and budget management. Additionally, it provides risk identification and comparison of different project alternatives. This section describes the COCOMO estimation for PawPal.

8.1. What is COCOMO?

COCOMO, which stands for COnstructive COst MOdel, is a procedural cost estimation model proposed by Barry Boehm in the late 1970s. The model is algorithmic and empirical in nature, based on data from numerous software projects. It uses regression formulas to estimate the required effort (person-months), development time (months), and cost, considering parameters such as the size of the system, product, project, and team factors.

COCOMO's main goal is to estimate the effort and resources needed to develop a software system, given its size and other project characteristics. It helps project managers in understanding the time, resources, and costs involved, thereby aiding in better planning, budget allocation, and risk management.

8.2. Types of COCOMO

COCOMO divides software projects into three categories based on their size and complexity:

- **Organic Project:** These projects are small, simple, and handled by small teams with good domain knowledge. They usually have a few rigid requirements. Example: Simple business systems, small inventory management systems, or data processing systems.
- **Semi-detached Project:** These projects are of intermediate size and complexity, with teams consisting of both experienced and inexperienced members. The project requirements can be either rigid or flexible. Example: Database design, operating system development.
- **Embedded Project:** These projects have high complexity, large team sizes, and multiple departments (hardware, software, operations). Example: Air traffic control, traffic light control systems.

Given that **PawPal** has a team of developers with good domain knowledge, and the project's requirements are rigid but not overly complex, **PawPal** is categorized as an **Organic Project**.

8.3 COCOMO Calculation for PawPal

There are three types of COCOMO models that can be applied to software projects:

1. **Basic COCOMO:** This model is the simplest and quickest, using the estimated size of the software (in Kilo Lines of Code, KLOC) as the primary input. Based on the type of project (Organic, Semi-detached, or Embedded), it uses predefined constant multipliers to calculate effort (person-months) and development time (months).
2. **Intermediate COCOMO:** This model builds on the Basic COCOMO by considering related cost drivers. These cost drivers include product parameters, hardware parameters, resource parameters, and project parameters, influencing the overall effort and schedule of the project.
3. **Detailed COCOMO:** The most comprehensive version, this model looks deeper into the project by breaking down the effort into development phases such as requirements engineering, design, coding, testing, etc.

For PawPal, we will use the Basic COCOMO model due to the **Organic Nature** of the project, its rigid requirements, and the need for a simple, quick, rough estimation. Given that PawPal is a mobile application, KLOC will primarily drive the estimation process.

8.4. Basic COCOMO Formula

The Basic COCOMO model uses the following formula:

- **Effort (E) = a × (KLOC)^b**

- **Development Time (D) = c × (E)^d**
- **Staff Size (S) = E / D**

Where:

- **KLOC** is the estimated number of thousands of lines of code.
- **a**, **b**, **c**, and **d** are constants based on the project type (organic, semi-detached, or embedded).

Project Type	a	b	c	d
Organic	2.40	1.05	2.50	.38
Semi-detached	3.00	1.12	2.50	.35
Embedded	3.60	1.20	2.50	.32

For an Organic Project, the constants are typically:

- **a = 2.4**
- **b = 1.05**
- **c = 2.5**
- **d = 0.38**

Using these constants, we can calculate the **Effort**, **Development Time**, and **Staff Size** for **PawPal**.

8.5. Applying COCOMO to PawPal

PawPal is a web-based application, the codebase is written in JavaScript. According to the Quantitative Software Management (QSM), the lines of code per function point is 47 (Software Estimation Resources | QSM). So,

for PawPal,

LOC (Line Of Code) = 47 × 444.28 = 20881.16 or 20.88116 kilo lines of code (KLOC).

8.6. Effort Calculation

In basic COCOMO, Effort (E) is calculated using the equation:

Effort, $E = a \times (KLOC)^b$

Where:

- **$a=2.40$**
- **$b=1.05$**
- **$KLOC=20.88116$**

So,

$E = 2.40 \times (20.88116)^{1.05} = 58.338$

**For PawPal, the effort required is 58.338 Person/Months. **

8.7. Development Time

In basic COCOMO, Development Time (D) is calculated using the equation:

De*velopment Time*, $D = c \times (E)^d$

Where:

- **$c=2.50$**
- **$d=0.38$**
- **$E=58.338$**

So,

$D=2.50 \times (58.338)^{0.38} = 11.722$

For PawPal, the development time is 11.722 Months.

8.8. People Required

In basic COCOMO, Staff Size (people required) is calculated using the equation:

Staff Size = Effort (E)/Development Time (D)

Where:

- **Effort (E) = 58.338**
- **Development Time (D) = 11.722**

So,

Staff Size = 58.338/11.722 = 4.976 ~ 5.00

For PawPal, the staff size is 5.00 People.

8.9. Productivity

In basic COCOMO, Productivity (in terms of KLOC per person-month) is calculated using the equation:

Productivity = KLOC/Effort (E)

Where,

- **KLOC = 20.88116**
- **Effort (E) = 58.338**

So,

Productivity = 20.88116/58.338 = 0.35

For PawPal, the productivity is 0.35 KLOC/Person-Month.

9. Gantt Chart

Complex or long-term projects can be overwhelming. Breaking a task into smaller subtasks can help it feel more manageable, and backwards planning has been shown to increase motivation to complete a project . In addition to making the project easier, learning how to break down a project into smaller chunks, to delegate tasks within a group, and schedule a timeline for successful execution of a project will be invaluable to any future career.

One method to manage projects this way is called a Gantt Chart.

9.1. What is a Gantt Chart?

A Gantt Chart is a visual project management timeline. It consists of horizontal bars, usually color coded, that show all the tasks of a project, when they are due, who is responsible, and how long each task will take.

9.2. Task Breakdown with Time, Start & Due date

Project: PawPal - Your Pet’s Digital Companion **Due: 15.05.2025**

Serial no.	Task	Time	Start date	Due date
1	Requirements Analysis	80 hour	2 February,2025	16 February,2025
2	System Design	80 hour	16 February,2025	2 March,2025
3	User/Admin & Profiles	54 hour	2 March,2025	8 March,2025
4	MediPaws & PawCessories	54 hour	8 March,2025	16 March,2025
5	VetConnect & Appointments	54 hour	16 March,2025	23 March,2025
6	PawMeet Live	40 hour	23 March,2025	1 April,2025
7	PawSitter Service	80 hour	1 April,2025	15 April,2025
8	PawTalk Forum	40 hour	15 April,2025	22 April,2025
9	VetPulse & Notifications	36 hour	22 April,2025	24 April,2025
10	PawAlert (Emergencies)	40 hour	24 April,2025	1 May,2025
11	Testing & QA	48 hour	2 May,2025	7 May,2025
12	Deployment	48 hour	8 May,2025	15 May,2025

9.3. Filling the Gantt Chart

Project: PawPal - Your Pet's Digital Companion **Due: 15.05.2025**

Task / Date->	2/2	16/2	2/3	8/3	16/3	23/3	1/4	15/4	22/4	24/4	1/5	2/5	7/5	8/5	15/5
Requirements Analysis	x	x													
System Design		x	x												
User/Admin & Profiles			x	x											
MediPaws & PawCessories				x	x										
VetConnect & Appointments					x	x									
PawMeet Live						x	x								
PawSitter Service							x	x							
PawTalk Forum								x	x						
VetPulse & Notifications									x	x					
PawAlert (Emergencies)										x	x				
Testing & QA												x	x		
Deployment														x	x

9.4. Individual Contribution

Md. Rakibur Rahman Lipu - (Abstract, Introduction, Vision & Goal, Planning, WBS, Wideband Delphi Estimation, Cocomo, Function Pointer, Conclusion)

Abdullah Abrar Labib - (Abstract, Introduction, Vision & Goal, Planning, WBS, Wideband Delphi Estimation, Cocomo, Gantt Chart, Conclusion)

Khan Israk Ahmed - (Abstract, Introduction, Vision & Goal, Planning, WBS, Wideband Delphi Estimation, Cocomo, Gantt Chart, Conclusion)

Zawed Bishwas - (Abstract, Introduction, Vision & Goal, Planning, WBS, Wideband Delphi Estimation, Cocomo, Function Pointer, Gantt Chart, Conclusion)

Maharab Hossain - (Abstract, Introduction, Vision & Goal, Planning, WBS, Wideband Delphi Estimation, Cocomo, Function Pointer, Conclusion)

10. Conclusion

PawPal is a thoughtfully built, all-encompassing digital ecosystem that attempts to meet the unique and expanding demands of pet owners in Bangladesh and elsewhere. Throughout the investigation, it becomes clear that PawPal fills a vital gap between traditional, fragmented pet-care services and a modern, user-centric platform. PawPal enables pet owners to manage all aspects of their pet's well-being with ease, convenience, and compassion by combining veterinary care, teleconsultation, emergency support, pet-sitting, grooming, and a dedicated community forum in a single, multilingual web application.

The development roadmap, which is backed by Agile methodology and an incremental delivery strategy, guarantees that each product is designed with quality and usability in mind. Detailed estimations, including task breakdowns, Gantt charts, function point analysis, and COCOMO modeling, demonstrate the project's feasibility, realistic timelines (~12 weeks/3 months), and doable team size (~5 persons). Technical complexity and potential concerns, such as data privacy, security, and scalability, have been thoroughly considered and handled via deliberate architecture and regular testing.

PawPal is not only functional and trustworthy, but also easily accessible, thanks to regionalized solutions such as Bengali language support and bKash/Nagad integrations. The incorporation of modern technologies such as AI-powered symptom analysis and geolocation enhances its value proposition and impact. Beyond its technology capabilities, PawPal promotes a lively and ethical pet-owner community by allowing members to connect, share expertise, and support one another.

In short, PawPal is a forward-thinking, well-planned, and socially responsible service that will make pet care more compassionate, seamless, and widely accessible. It lays the groundwork for future additions, such as wearable integrations and mobile apps, and serves as a paradigm for scalable, regionally tailored pet-care systems. PawPal will make sure that no pet goes uncared for and that no pet owner feels alone.