Traditional spam filters and rule-based security systems **fail to detect advanced phishing attacks** due to their static nature. Therefore, a dynamic, learning-based approach is required to improve detection accuracy.

2.2 Empathy Map Canvas

Understanding the concerns and pain points of financial users is crucial for developing an effective phishing detection system.

- **What users think and feel:** Users are worried about financial loss and identity theft. They may feel insecure about opening emails from unknown sources.
- **What users see:** They often encounter emails with urgent warnings, fake security alerts, or offers that appear too good to be true.
- **What users hear:** People may receive phishing emails pretending to be from banks, government agencies, or well-known companies.
- **What users say and do:** Some users may unknowingly click on phishing links, while others may report suspicious emails to IT departments.

2.3 Brainstorming

Several approaches were considered for detecting phishing emails:

1. **Rule-based Detection:** Uses predefined rules to classify emails, but it may not detect new phishing techniques.
2. **Machine Learning Models:** Analyzes email content, sender details, and metadata to classify emails with high accuracy.
3. **Deep Learning Approaches:** Utilizes neural networks to detect complex phishing patterns.

After evaluating these methods, a **machine learning-based approach** was selected as it offers better adaptability and accuracy.

---

3. REQUIREMENT ANALYSIS

3.1 Customer Journey Map

The phishing detection system follows these steps:

1. **User receives an email** – The system captures incoming emails and prepares them for analysis.
2. **System scans and analyzes email features** – The email's subject, content, sender, and embedded links are examined.
3. **Machine learning model predicts legitimacy** – The trained model classifies the email as phishing or legitimate.
4. **User receives a warning if the email is phishing** – A security alert is generated for suspicious emails.

3.2 Solution Requirement

To develop an efficient phishing detection system, the following components are required:

- **Data Preprocessing Pipeline** – Cleans and structures the email data for analysis.
- **Machine Learning Model** – A classifier trained on phishing and legitimate email datasets.
- **Performance Evaluation Framework** – Assesses accuracy, precision, recall, and F1-score.
- **User Notification System** – Alerts users about detected phishing emails.
- **Deployment Infrastructure** – A web-based or API-driven system for real-time email classification.

3.3 Data Flow Diagram

A structured flow from data collection to model training and deployment:

1. **Data Collection** – Emails are gathered from phishing datasets and legitimate sources.
2. **Data Preprocessing** – Tokenization, stopword removal, and text cleaning are performed.
3. **Feature Engineering** – Extracts key indicators such as URLs, email headers, and NLP-based text analysis.
4. **Model Training & Evaluation** – Various ML models are trained and tested for accuracy.
5. **Deployment & Alert System** – The system is deployed, and phishing alerts are sent to users.

3.4 Technology Stack

The following technologies and frameworks are used:

- **Programming Language:** Python
- **Libraries:** Scikit-learn, Pandas, NumPy
- **Natural Language Processing (NLP):** Tokenization, TF-IDF, word embeddings
- **Web Frameworks:** Flask or Django for deployment
- **Database:** MySQL or PostgreSQL
- **Cloud Services (Optional):** AWS, Google Cloud for scalable deployment

---