

The slide features a light gray background with abstract geometric shapes in blue and dark blue at the top and bottom. On the left side, there are several thin, dark gray lines that resemble circuit traces, some ending in small circles. On the right side, there are similar circuit-like lines, including a horizontal line with a small circle at its end.

Selling app Based of Temmu

Jefferson David Rico Ruiz

Nelson Navarro de la Rosa

TABLE OF CONTENTS

01

Introduction

02

Requirements

03

**Designing User
Stories**

04

**Designing of
Mockups**

05

**Raises the CRC
cars**

06

**Package
organization**

TABLE OF CONTENTS

07

UML Diagrams

08

**Solid
Implementation**





01 Introduction



What is our project about?

- A product buying and selling app

What are our goals?

- Develop a product buying and selling app
- Simulated a purchasing process
- Connects independent sellers with buyers



Why did we choose to do this project?

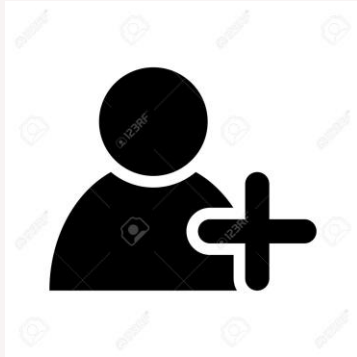


- The change in the way we buy in recent years
- Technological advances
- Changes in consumer habits



02 Requirements

Functionals



- User Registration: The user must be able to register
- Product Management :The "Seller" must be able to upload products
- Buy product: The user must be able to purchase the desired product.

Non-Functionals



- An interactive and easy-to-use interface
- Security in the data
- Performance: respond efficiently to user interaction



03 **Designing User Stories**

TITLE: Main interface
(Customer)

PRIORITY: High

ESTIMATE: 20-24 hours

User Story: As an app user, I want to have a main section that allows me to easily access the products I need through categories.

Acceptance Criteria: Given a series of product categories, when the user interacts with them, he or she will be able to access the section of the selected category.

TITLE: Product Purchase Interface
PRIORITY: High

ESTIMATE: 16-20 hours

User Story: As a user, I want to be redirected to a section that displays the product image, price, description, user reviews, and a "Buy Now" button . This allows the user to choose the product they want to purchase and obtain more information.

Acceptance Criteria: Once a product is selected, a section will be displayed with the product and its information.

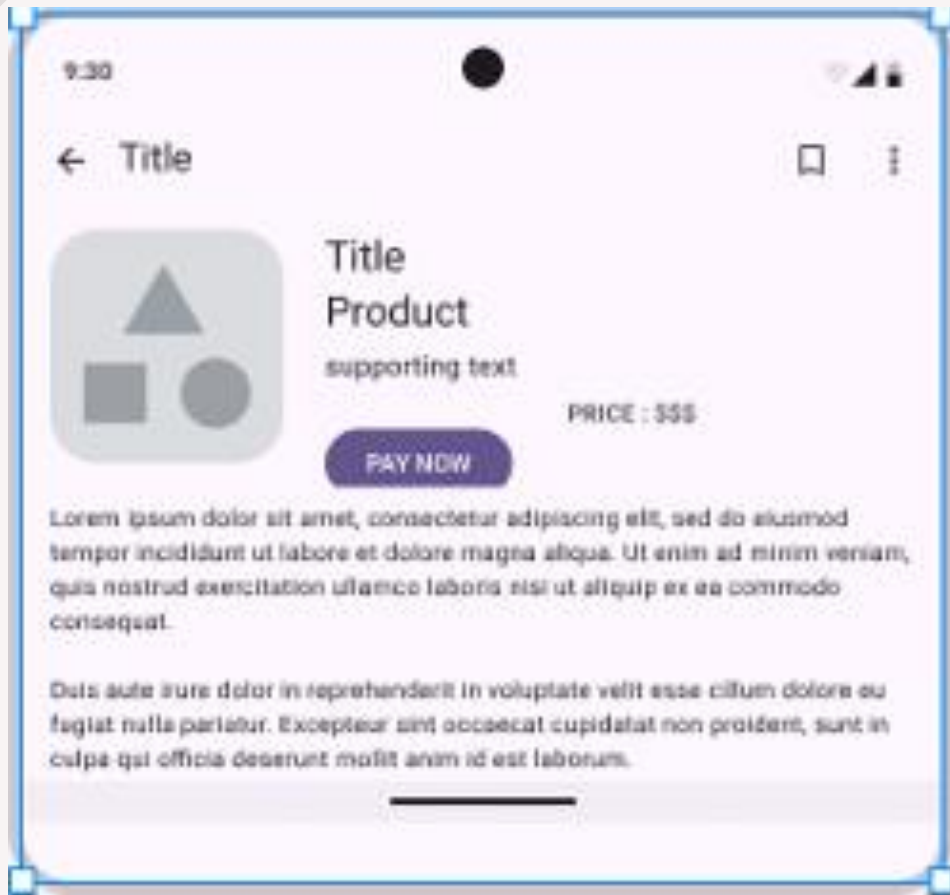
TITLE: Add Product **PRIORITY:** Medium **ESTIMATE:** 16-20 hours
Interface

User Story: As a seller, I want a space where I can add the product to the application and enter data such as the name, price, image, and description, and assign it to the category to put the product for sale.

Acceptance Criteria: Given the add product button, when the seller interacts with it, they should be redirected to that section.



04 **Designing Mockups**




Mockup Selected Product



Mockup Main Interface Customer

Details



Name of product

Price

Description of product

Categorie of product

Assign stock

Publish product

Mockup Add Product



05 **Designing CRC Cards**

Class: Product

Responsibilities

- Have a photo, description, and price to provide information to the buyer
- Know what type of item it is and, therefore, what category it should be associated with
- Going on offer eventually

Collaborators

Seller

- This person creates and manages product features. They can implement offers, modify prices, and modify images

Customer

- The customer can view and purchase the product

Payment

- The product can pass data to the Payment class, such as its price and image

Offer

- The offer selects a product

CRC CARDS

Class: Customer

Responsibilities

- Order the product and finish paying for it
- Have the ability to choose the desired product(s) across categories
- Provide information when paying for a product

Collaborators

Product

- You can view and buy many products, each from different sellers.

Payment

- The customer can use this class to make a purchase

Profile

- The customer can enter login information such as their name, age, etc. to validate each one

Offer

- The customer can interact with the displayed offer

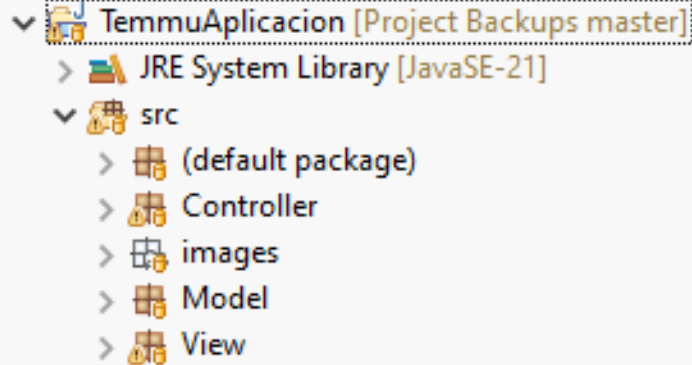


06 **Directory Package**

What does it consist

4 Folders: Controller, Model, images, and View

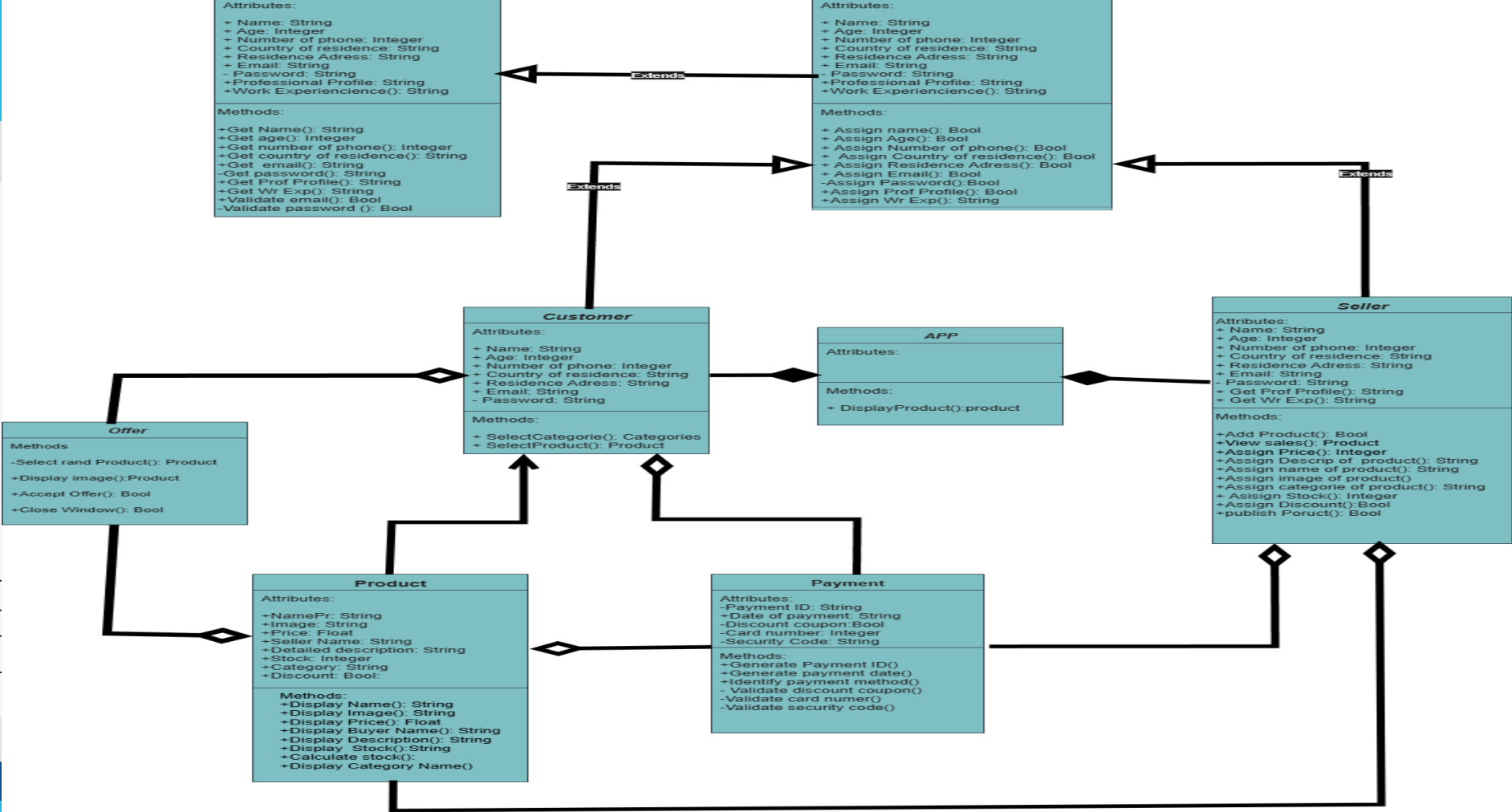
- DefaultPackage: Contains the main class
- Controller: Data validation classes
- View: Contains the graphical interfaces
- Model: Controls reading and writing data in text files
- images: Contains images and text files
- Follow the MVC pattern



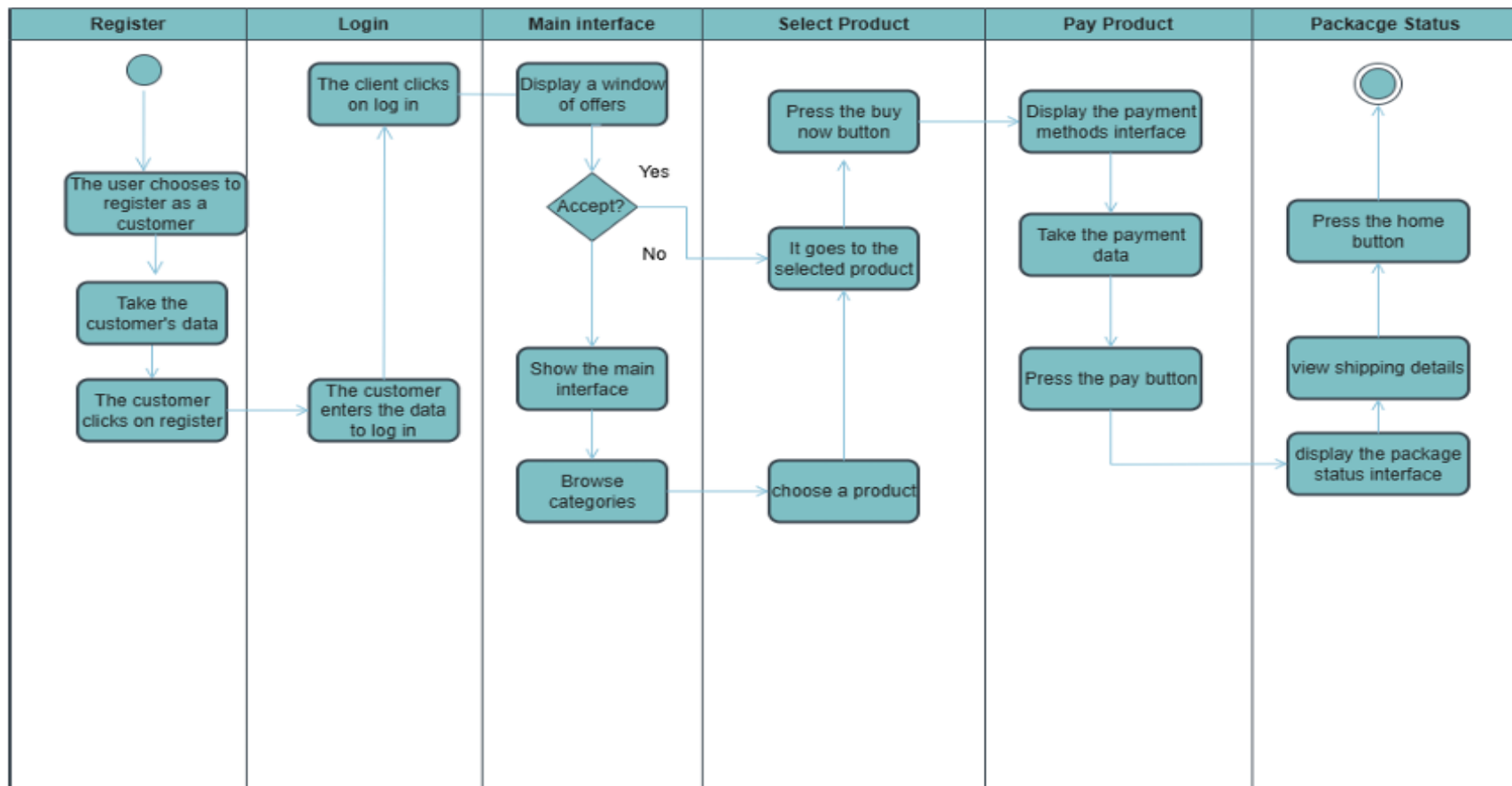


07 **UML Diagrams**

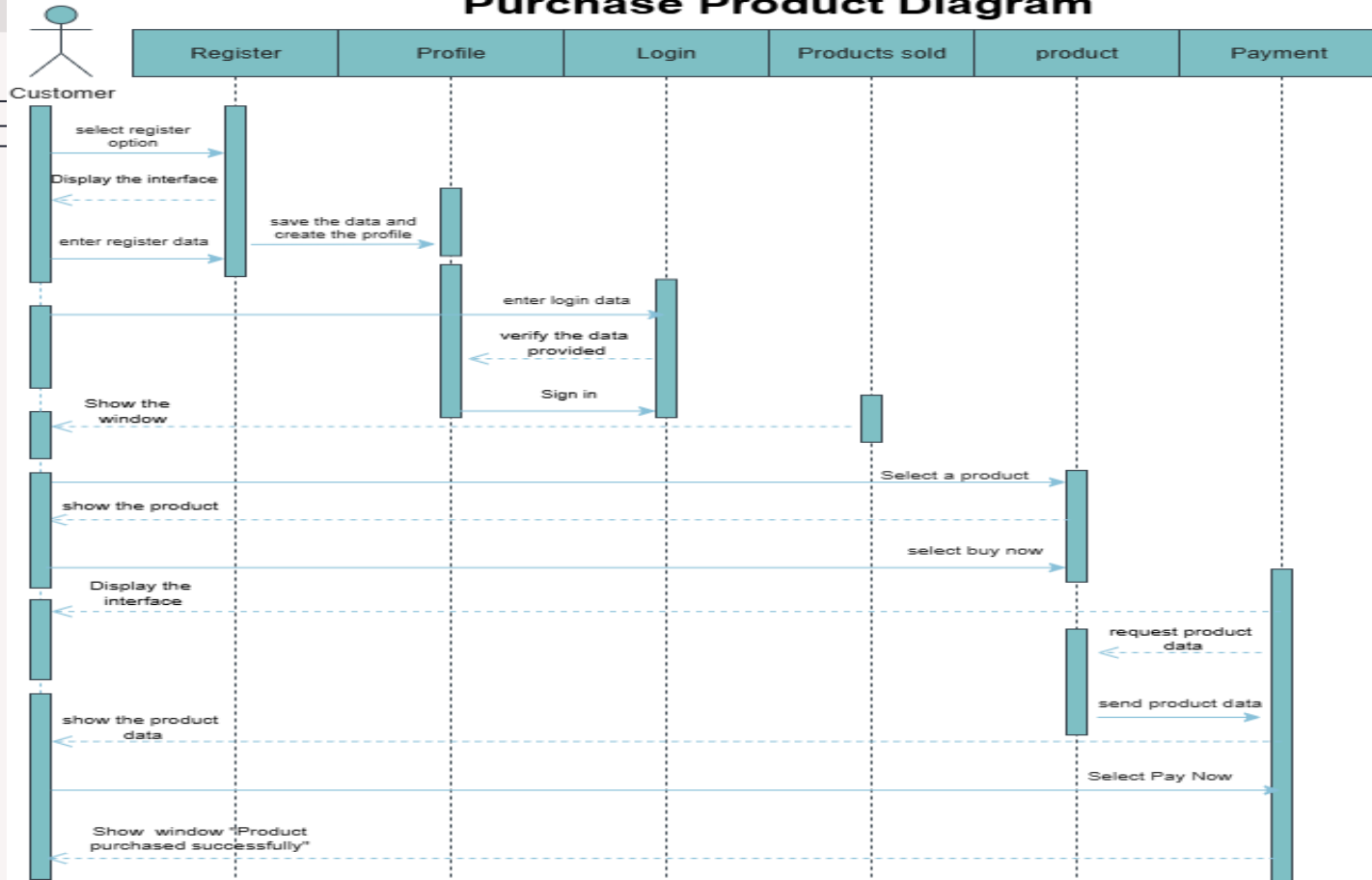




Activity Diagram Product Purchase



Purchase Product Diagram





08 **Solid Implementation**

1. Single responsibility: The Product class only handles product data, and the Registration class only handles user data.

2. Open to extension, closed to modification: The Seller and Customer classes can be allowed to see their data.

3. Liskov Substitution: No low-level classes were missing.

4. Interface Segregation: No interfaces were missing.

5. Dependency Inversion: No low-level modules were missing.

S.O.L.I.D.

