**Software Testing Assignment**

**Module–1(Fundamental)**

**• What is SDLC**

* **SDLC is a software development life cycle. SDLC is a structure imposed on the development of a software product defines the process for planning, implementation, testing, documentation, deployment, ongoing maintenance and support.**

**• What is software testing?**

* **Software testing is a process used to identify the completeness,correctness, and quality of developed computer software.**

**• What is agile methodology?**

* **This method is Currently used in market. The Agile methodology is combination of iterative and incremental model and is a project management approach that involves breaking the project into phases and emphasizes continuous collaboration and improvement.**

**• What is SRS**

* **A Software requirements specification (SRS) is document that describe what the software will do and how it will be expected to perform. It is the main point of reference for the entire team.**
* **Types of requirements:**

**1.CUSTOMER REQUIREMENTS**

**2.FUNCTIONAL REQUIREMENTS**

**3.NON-FUNCTIONAL REQUIREMENTS**

**• What is oops**

* **OOPS: Object Oriented Programming System**
* **In this Black Box Testing, Functional Testing**
* **OOP is based on the concept of objects. Objects are defined each with its own properties or attributes.**

**• Write Basic Concepts of oops**

* **OBJECT**
* **CLASS**
* **ENCAPSULATION**
* **INHARITANCE**
* **POLYMORPHISM**
* **ABSTRACTION**

**• What is object**

* **Instances of an class**
* **Its same as class member**
* **Using new keyword and constructor through create object**

**• What is class**

* **Is an collection of data member (variables) and member function (method, process) with its behavior**

**• What is encapsulation**

* **Wrapping up of data into single unit i.e. :data hiding purpose**

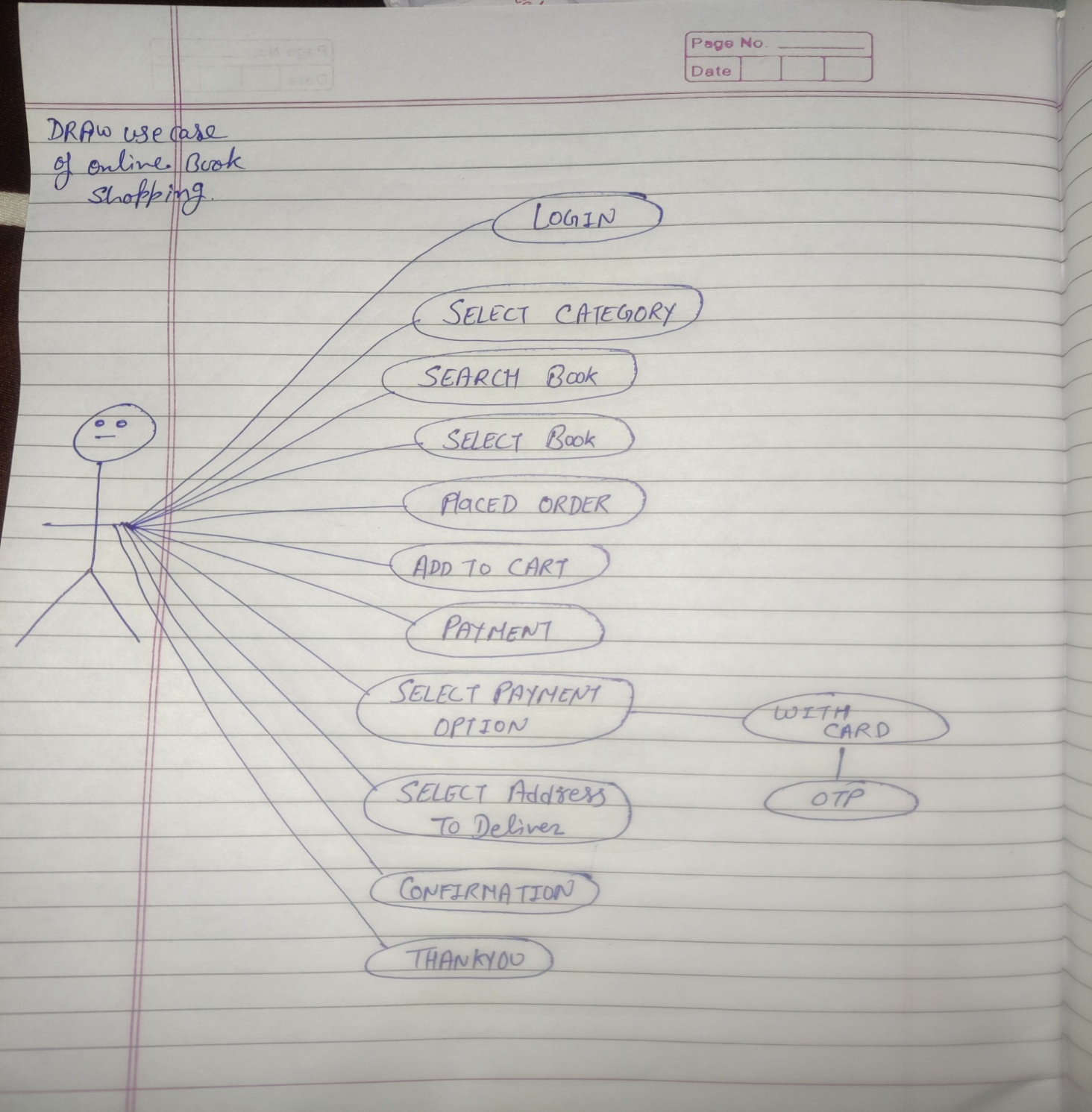
**• What is inheritance**

* **Properties of parent class extends into child class**
* **Main purpose is reusability, extensibility**
* **There are mainly 5 types of**
* **1.SINGLE : only one parent having only one child**
* **2.MULTILEVEL: single inheritance having one more another child**
* **3.HIERARCHICAL: one parent having 2 or more child**
* **4.MULTIPLE: java does not support directly**
* **5.HYBRID: java does not support directly**

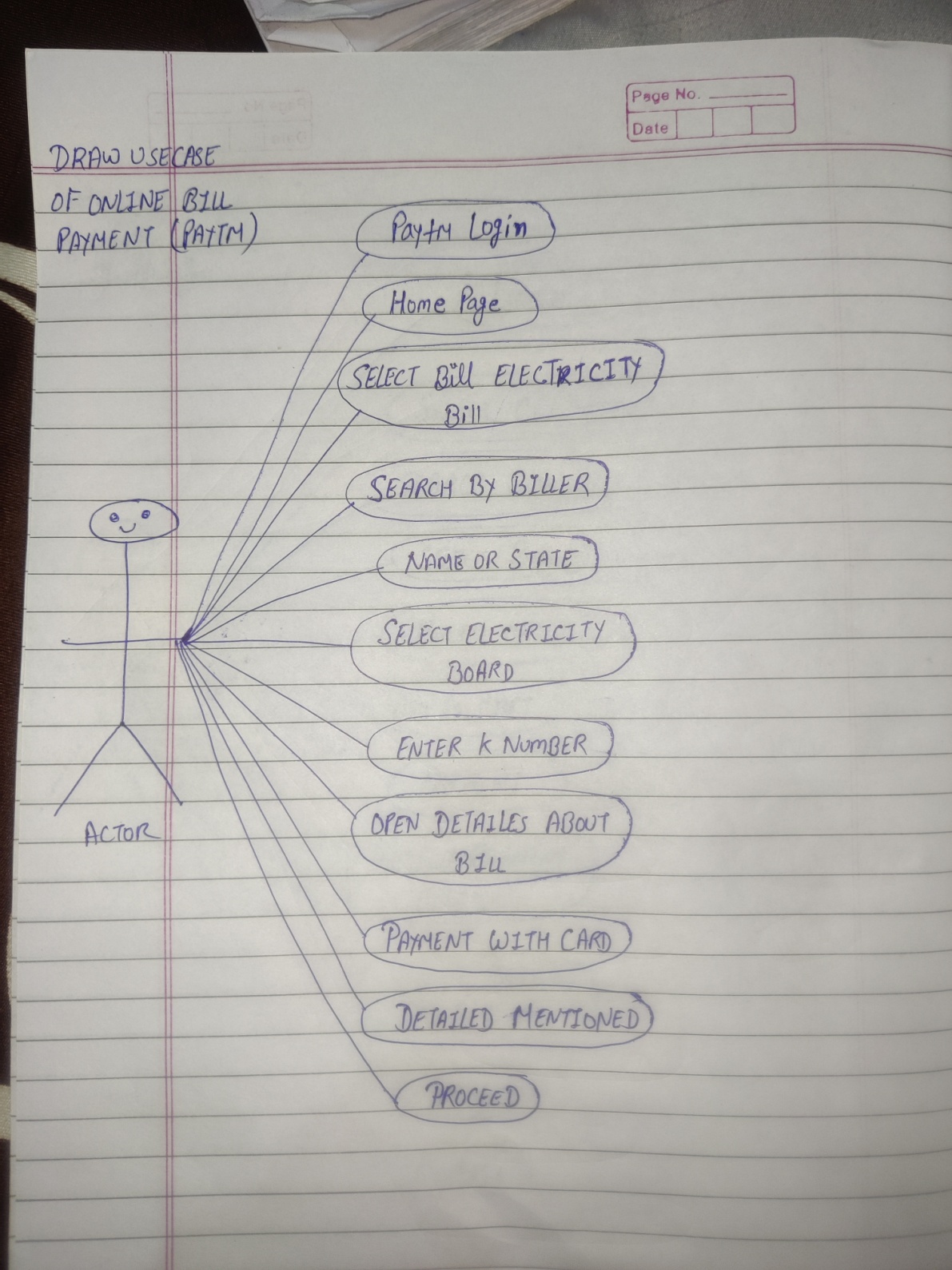
**• What is polymorphism**

* **Ability to take one name having many forms or multiple forms**
* **There are 2 types:**
* **1.METHOD OVERLOADING (Compile time)**
* **2.METHOD OVERRIDING (Run time)**
* **What is Abstraction**
* **only essential part should be display rest of the part will be hide**
* **Data hiding in extended level**

**• Draw Usecase on Online book shopping**



**• Draw Usecase on online bill payment system (paytm)**



**• Write SDLC phases with basic introduction**

* **Requirement Collection/Gathering: Establish customer needs. Requirement definitions usually consist of natural language, supplemented by diagrams and tables.**

**Three types of problem scan arise:-**

* **Analysis : 1. Lack of clarity 2. Requirement confusion 3. Requirement amalgamation. Its define the requirement of the system, independent of how these requirement will be accomplished.**
* **This analysis represents the “what” phase.**
* **This phase represents the “how” phases.**
* **Desing Phase: Desing Architecture document, Implementation plan, Critical priority analysis, Performance analysis, Test plan.**
* **Implementation Phase: In this phase, the team builds the components either from scratch or by the composition.**
* **Implementation-code**
* **Critical error Removal**
* **Testing Phase: Simply stated, Quality is very important many companies have not learned about importance of quality and deliver more claimed functionality but at a low level of quality.**
* **Internal testing, regression testing, unit testing, application testing, stress testing.**
* **Maintenance phase: In this phase, maintenance comes after deployment of the software into the field.**
* **Configuration and version management**
* **Reengineering( Redesigning and Refactoring)**
* **Updating all analysis, design and user documentation.**

**Three types of maintenance phase which are:**

**1. Corrective Maintenance 2. Adapting Maintenance 3. Perfective Maintenance.**

**• Explain Phases of the waterfall model**

* **This model is step by step model. In this every phases are sequentially. We not go back and can’t change in it.**
* **Requirements must be “Frozen” to early in the life cycle.**
* **Requirement are validated too late.**
* **Applications(When to use)**
* **Requirements are well documented, clear and fixed.**
* **Product Definition stable.**
* **There are no ambiguous requirements.**
* **The project is short.**
* **Pros(Why waterfall model):**
* **Simple and easy to use and understand.**
* **Easy to manage.**
* **Clearly defined stages.**
* **Well understood milestone.**
* **Easy to arrange tasks.**
* **Process and result are well documented.**
* **Cons(why not waterfall model):**
* **High amount of risk and uncertainty.**
* **Not good of complex and object oriented projects.**
* **Poor model for long and ongoing projects.**

**• Write phases of spiral model**

* **Its known as Bhoem’s Spiral model. Spiral model was widely use in the software industry. It is similar to the incremental development for a system with more emphasis placed on risk analysis.**
* **Four phases are:**
* **1.Planing**
* **2.Risk analysis**
* **3.Engineering (Development)**
* **4.Customer evaluation**

**• Write agile manifesto principles**

* **Agile manifesto are:**
* **1. Individual interaction**
* **2. Working software**
* **3. Customer collaboration**
* **4. Responding to change**

**• Explain working methodology of agile model and also write pros and cons.**

* **Agile model, In agile tasks are divide into time boxes(small time frames) to deliver specific features for a release. In this no deadline concept used.**

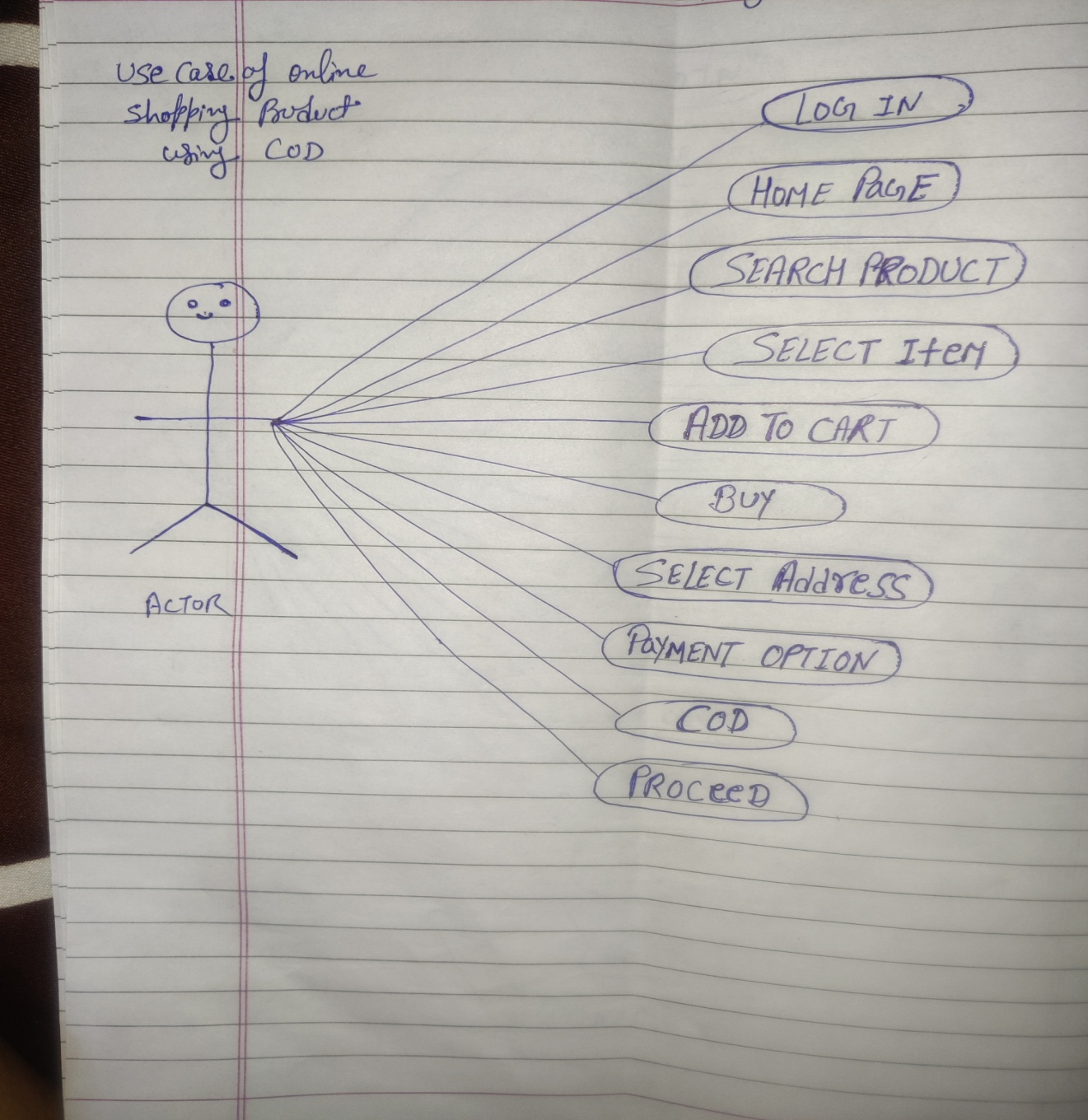
**Pros:**

* **realistic approach to software development**
* **functionality can be developed rapidly**
* **suitable for fixed or changing requirements**
* **easy to manage**
* **gives flexibility to developers**

**Cons:**

* **not suitable for handling complex dependencies**
* **more risk of sustainability, maintainability, extensibility**
* **heavily depends on customer interaction so if customer is not clear team can move in the wrong direction**
* **transfer of technology to the new team member is quite challenging due to lack of documentation**

**• Draw usecase on Online shopping product using COD.**



**• Draw usecase on Online shopping product using payment gateway**

