Midterm Exam

St. Name: Kamble, Soham
Last Name, First Name

Important Notes:

- There are 5 questions in the exam paper
- Type your answers (**NO hand-writing**) below every question.

Q1. (10 points) Compare and contrast the traditional waterfall method and agile method for engineering software products.
Ans.

The waterfall model is a linear and sequential approach where each phase of the software development life cycle must be completed before the next phase begins. It's structured, systematic, and follows a top-down approach, beginning from system and software requirements, moving on to design, coding, testing, and eventually, maintenance. If project outcomes are clearly understood from the start and there's a need to meet strict regulations, Waterfall can be the most suitable because it requires specific deliverables for each phase before progressing. Conversely, the Agile methodology promotes iterative development, where the product is built incrementally from the start, in small sections developed and tested in cycles. Agile is better suited for teams aiming to move swiftly, experiment with their direction, and don't have a solid vision of the end product before commencing.

In the waterfall model, changes are typically discouraged once requirements are locked down in the early stages. Any alterations later in the process can be costly and can significantly delay the project. Agile, on the other hand, embraces change, readily accommodating modifications even late in the development cycle due to its short sprints. Agile's inherent flexibility is ideal for teams that anticipate shifts in project direction, requiring a cooperative, self-motivated team and frequent interactions with business owners and stakeholders for feedback.

Waterfall projects often gather requirements from stakeholders initially and then the development team might work in relative isolation until the final product's delivery. In contrast Agile emphasized continuous collaboration between the development team and stakeholders. Stakeholders, including customers and end-users, engage at regular intervals to review and prioritize features, ensuring that the developing software is aligned with evolving user needs and business goals throughout it's creation.

The choice between these methodologies should be dictated by the specific project's demands, the team's dynamics, and the anticipated changes during development process.

Q2. (10 points) Compare and contrast Function Points, Source Line Of Code (SLOC), and Use-Cases (UC) for measuring software size.

Ans.

Function Points (FP), Source Lines of Code (SLOC), and Use-cases (UC) are prominent metrics in the software industry, each giving a unique lens through which one can understand the size or complexity of a software system. Function Points are derived from the user's perspective of the software, focusing on its inherent functionalities. They are measure based on multiple factors such as the system's inputs, outputs, inquiries, interfaces, and data files. The primary advantage of FP is its emphasis on what the user values, making it possible to compare across different projects and technologies. However, it comes with its set of challenges. To get a precise FP count, one requires a specialized skill set. Also, given its high-level nature, it might overlook some granular complexities present in the code.

SLOC, on the other hand, is a more tangible metric as it directly counts the lines present in the source code, excluding and comments and white spaces. It offers an unambiguous way to gauge productivity, for instance, by evaluation the number of defects per line of code. However, its applicability is somewhat limited by its dependence on the programming language. For example, what can be achieved with 1000 lines in one language might differ significantly in another. A higher count of lines doesn't automatically translate to better quality, and oftentimes, succinct code is more efficient and elegant. Use-Cases delve into the interactions between the system and its users. By mapping out specific scenarios of usage, they provide insights into the system's size. Their strength lies in their user-centric approach which is not just crucial for estimating size and complexity but is also invaluable for gathering requirements. However, they aren't without their pitfalls. UCs might sometimes overlook non-functional requirements, and the intricacies that come with actual implementation can be sidelined. Moreover, its subjective nature might lead to inconsistencies, as two developers might perceive and jot down different use-cases for identical functionalities.

In conclusion, when we talk about measuring software's size or complexity, there isn't a one-size-fits-all metric. Whether it's FP's user-centric, language-agnostic view, SLOC's direct peek into the code, or UC's emphasis on user interactions, each has its own strengths and limitation. The best approach hinges on the specifics of the project and what one hopes to glean from the measurements.

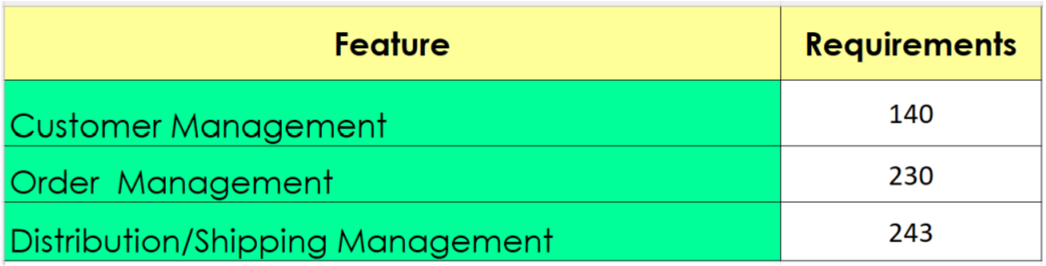
Q3. (10 points) Explain how the vision and scope document can be used in developing and planning the releases for the software product.

Ans.
The vision and scope document plays a pivotal role in the software development lifecycle. It serves to establish a clear understanding of the software product's objectives (vision) and the specific boundaries that the project intends to maintain (scope). The incorporation of this document into the development and planning od software releases can enhance the efficiency of the development process, set definitive priorities, and ensure that the end product aligns with the expectations of stakeholders.

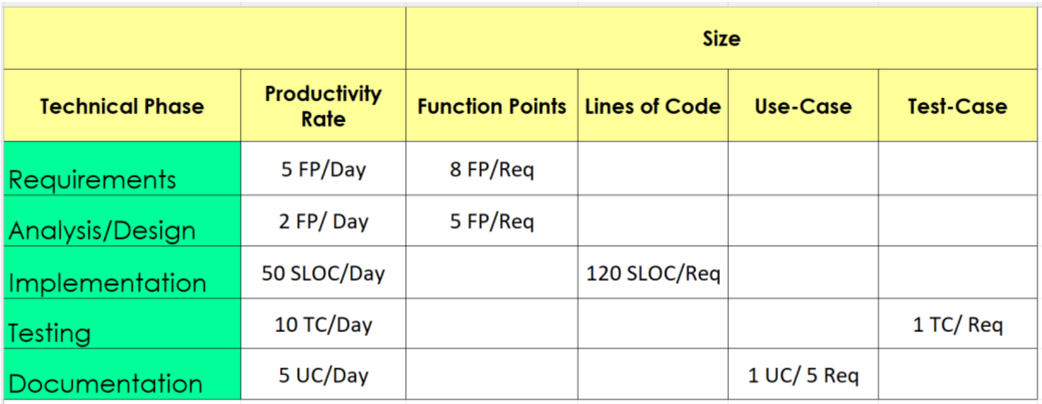
Within the vision and scope document, the vision section offers a broad perspective on the software's primary objectives, its intended audience, and the specific challenges it aims to address. This overarching understanding is instrumental in distinguishing between the features that should be introduced in the initial stages of release and those that can be scheduled for subsequent versions. On the other hand, the scope delineation ensures a clear distinction between essential product features and those which are outside the project's purview. This distinction between essential product features and those which are outside the project's purview. This distinction is pivotal for the process of feature prioritization, enabling teams to strategize across multiple release cycles and rank features according to their relevance to the project's vision and anticipated user impact.

The creation of this vision and scope document often involves collaboration with various stakeholders, ensuring their input and concerns are woven into the fabric of the project. This collaboration fosters alignment with stakeholder expectations, reducing the potential for unforeseen challenges and course adjustments as the project progresses. Moreover, the document offers other benefits, such as resource optimization, mitigation of scope creep, setting clear project expectations, and effective risk management. Ultimately, the vision and scope document acts as a beacon, guiding software development efforts. By utilizing it proficiently during the release planning phase, developers can ensure the software evolves in sync with its foundational objectives, stakeholder aspirations, and the ever-evolving demands of the market.

Q4. (30 points): Assume that the Retailer FastCleaningRobot has planned the following features for its first software release, calculate the total headcount effort needed to complete every phase for its software product development given the following measurements for the productivity of its engineers:



Feature	Requirements
Customer Management	140
Order Management	230
Distribution/Shipping Management	243



		Size			
Technical Phase	Productivity Rate	Function Points	Lines of Code	Use-Case	Test-Case
Requirements	5 FP/Day	8 FP/Req			
Analysis/Design	2 FP/ Day	5 FP/Req			
Implementation	50 SLOC/Day		120 SLOC/Req		
Testing	10 TC/Day				1 TC/ Req
Documentation	5 UC/Day			1 UC/ 5 Req	

Ans.

For each phase and feature:

Requirements:

Customer Management: Effort = 140 * 8 FP / (5 FP/Day) = 224 days / HC

Order Management: Effort = 230 * 8 FP / (5 FP/Day) = 368 days / HC

Distribution/ Shipping Management: Effort = 243 * 8 FP / (5 FP/Day) = 388.8 = 389 days / HC

Total = 981 days / HC

Analysis/ Design:

Customer Management: Effort = 140 * 5 FP / (2 FP/Day) = 350 days / HC

Order Management: Effort = 230 * 5 FP / (2 FP/Day) = 575 days / HC

Distribution/ Shipping Management: Effort = 243 * 5 FP / (2 FP/Day) = 607.5 = 608 days / HC

Total = 1533 days / HC

Implementation:

Customer Management: Effort = 140 * 120 SLOC / (50 SLOC/Day) = 336 days / HC

Order Management: Effort = 230 * 120 SLOC / (50 SLOC/Day) = 552 days / HC

Distribution/ Shipping Management: Effort = 243 * 120 SLOC / (50 SLOC/Day) = 583.2 = 584 days / HC

Total = 1472 days / HC

Testing:

Customer Management: Effort = 140 * 1 TC / (10 TC/Day) = 14 days / HC

Order Management: Effort = 230 * 1 TC / (10 TC/Day) = 23 days / HC

Distribution/ Shipping Management: Effort = 243 * 1 TC / (10 TC/Day) = 24.3 = 24 days / HC

Total = 61 days / HC

Documentation:

Customer Management: Effort = 140 * 1/5 UC / (5 UC/Day) = 5.6 = 6 days / HC

Order Management: Effort = 230 * 1/5 UC / (5 UC/Day) = 9.2 = 10 days / HC

Distribution/ Shipping Management: Effort = 243 * 1/5 UC / (5 UC/Day) = 9.72 = 10 days / HC

Total = 26 days / HC

Total Effort / HC = 981 + 1533 + 1472 + 61 + 26 = 4073 days / HC

Q5 . (40 points). **Chicobino's Pizza:** Consider the following overview and high-level descriptions for this system and create/document the items listed under the deliverables section.

Software Product Overview:

For its chain of fast food outlets, Chicobino's Pizza has recently installed a new online system to speed up deliveries and improve customer satisfaction. When a customer places an online order to have pizza delivered, an employee verifies the order and checks the customers phone number against a data store containing past orders. If the order is for a repeat customer, the system matches the number with the customer database and displays the customer record on the screen. For first-time customers, the employee (salesman) verifies the caller's name and address, and creates a record in the customer database.

The order taker (salesman) verifies the customer's pizza order within 10 minutes after the customer placed the order online. After 10 minutes from placing the order by the customer, the system prints out three-part order on a printer located in the kitchen. The original is used by the cook to prepare the order. When the order is ready, the chief marks the other two copies completed and gives them to the delivery driver or the order taker to serve as delivery receipts for the driver and customer, respectively. At the same time that the order is printed, the order taker's computer displays a city locator grid that is used to help dispatch the drivers.

From a copy of the display, a dispatch slip—showing the customer's street and connecting roads—is printed for the driver. The final system output generated at this time is a record of the order, which is the source for the event data written to the order system.

High-Level Description:

- Every store location has a StoreManager, Customers, and Salesmen, delivery drivers, chiefs
- The StoreManager can Add/Delete different types of pizzas offered
- The StoreManager can Add/Delete Chicobino coupons
- There are number of sports teams and each offers its own discount-coupons for its fans when buying pizza from Chicobino.
- Public schools can register with Chicobino and offer flyers for students families, where a public school collects 2% of the cost for every pizza bought by the student's family from Chicobino pizza
- Chicobino store offers the following types of pizzas:
    1. Cheese pizza

2. Veggie pizza with the choice of toppings:
    1. Mushrooms
    2. Green pepper
    3. Onions
    4. Spinach
    5. Olives
3. Meat pizza with the choice of meat
    1. Pepperoni
    2. Sausage
    3. Bacon

- The customer can choose one of the following delivery options when ordering the pizza
    1. Expedited delivery ($5 fee) – guaranteed delivery within 30 minutes
    2. Free delivery – guaranteed delivery within 75 minutes
- The customer can order any pizza online. However, there are a number of order options that the store likes to offer its customers
    1. Single order
    2. Double order – Two orders, one at 100% the actual price and the 2nd order expires in a week at 50% the actual price (Basically, it is buy one pizza get the 2nd one half-price)
    3. Triple order – Three orders, one at 100% the actual price , the 2nd order expires in a week at 50% the actual price, the 3nd order expires in a month at 50% the actual price (Basically, it is buy two pizzas get the 3rd one free)
- The customer can place an order and check the status of an order; every order has a unique order number
- The customer can cancel an order only within 10 minutes after placing the order
- The customer can update (addition of new pizzas only) an order if the order hasn't been loaded on the delivery truck yet
- The customer can pay in cash on delivery, or credit card online
- The customer has the choice to enroll (or cancel) and become VIP-Member in order to receive 10% discount for every pizza purchased for an annual fee of $100
- The store manager and order takers can query the system to know the current location for every pizza delivery driver/truck designated for the store location.
- For certain addresses, delivery drivers can't accept payment by cash on delivery
- The store manager and order takers can query the system to know the current number of pending orders on the waiting queue to be loaded into the delivery trucks when trucks are available (back from prior delivery)

Deliverables:

Given the Chicobino's pizza description listed above:

1. List the primary features.
2. List the primary user classes.
3. List 3 use-cases for every feature you identified above.
4. Create the Context diagram for Chicobino's pizza.
5. Create the Feature tree diagram Chicobino's pizza.

Ans.

1. Primary features:
    - **a)** Online Order System
    - **b)** Customer Database Management
    - **c)** Pizza Management System (Add/ Delete Pizzas & Toppings)
    - **d)** Coupon and Discount System
    - **e)** Delivery Dispatch System
    - **f)** Order Tracking and Management
    - **g)** Payment System
    - **h)** VIP-Member Management
2. Primary User Classes:
    - **a)** Customer
    - **b)** Salesman (Order Taker)
    - **c)** StoreManager
    - **d)** Chief (Cook)
    - **e)** Delivery Driver
3. Use-Cases for Every Feature:
    - **a)** Online Order System
        - **i.** Customer Places an order.
        - **ii.** Salesman verifies the order.
        - **iii.** Customer checks the status of an order.

    - **b)** Customer Database Management
        - **i.** Salesman verifies the customer's phone number for repeat customers.
        - **ii.** Salesman creates a new customer record.
        - **iii.** Salesman retrieves customer data.

    - **c)** Pizza Management System (Add/ Delete Pizzas & Toppings)
        - **i.** StoreManager adds a new type of pizza.
        - **ii.** StoreManager deletes a type of pizza.
        - **iii.** StoreManager updates pizza toppings.

    - **d)** Coupon and Discount System
        - **i.** StoreManager adds a new coupon.
        - **ii.** StoreManager deletes an existing coupon.

iii. Customer redeems a sports team or school coupon.

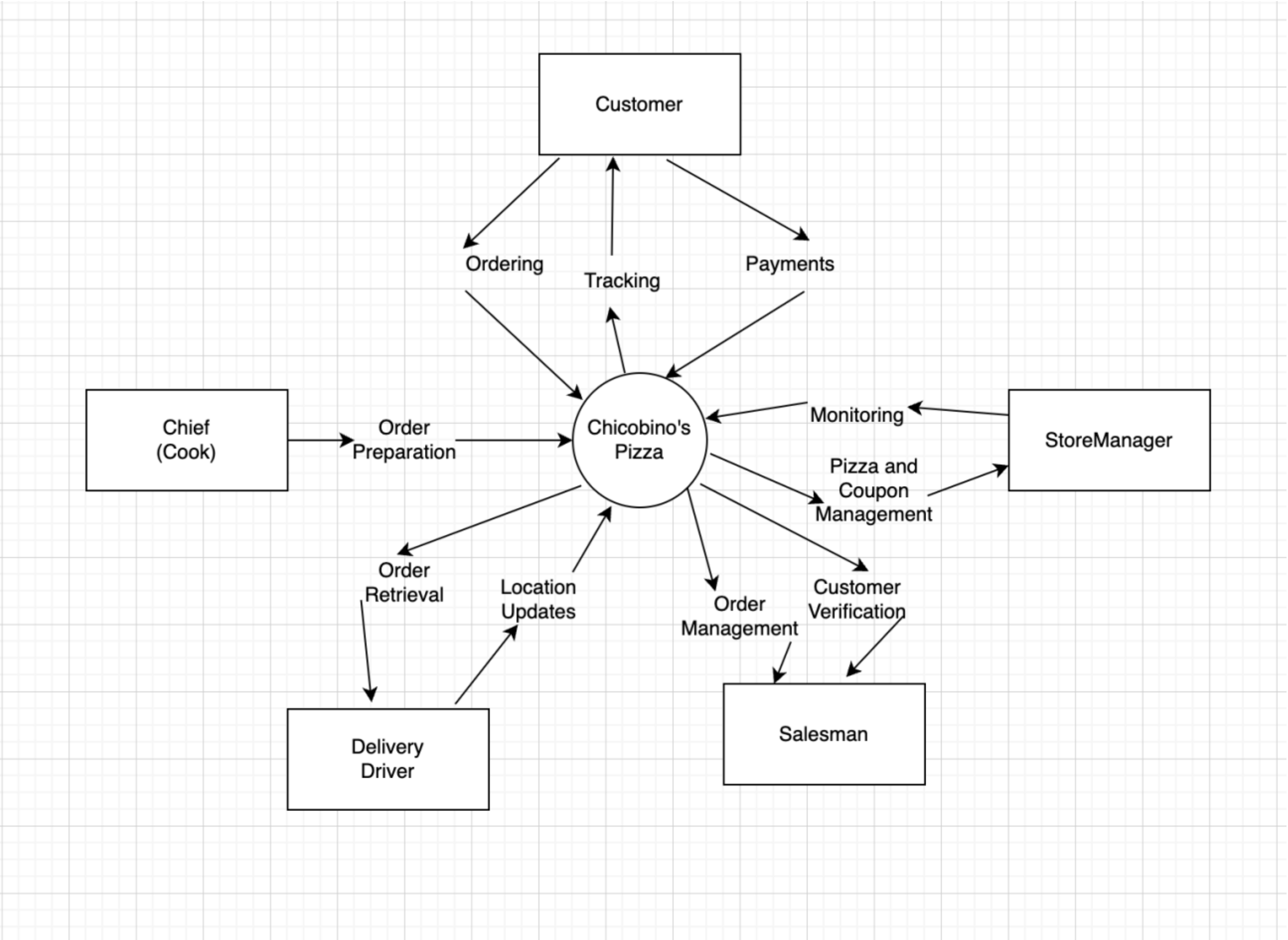
e) Delivery Dispatch System
   i. System displays a city locator grid for delivery.
   ii. Salesman prints dispatch slip for the driver.
   iii. Salesman queries the system for driver's location.

f) Order Tracking and Management
   i. Customer cancels the order.
   ii. Customer updates the order.
   iii. Salesman queries the system for pending orders.

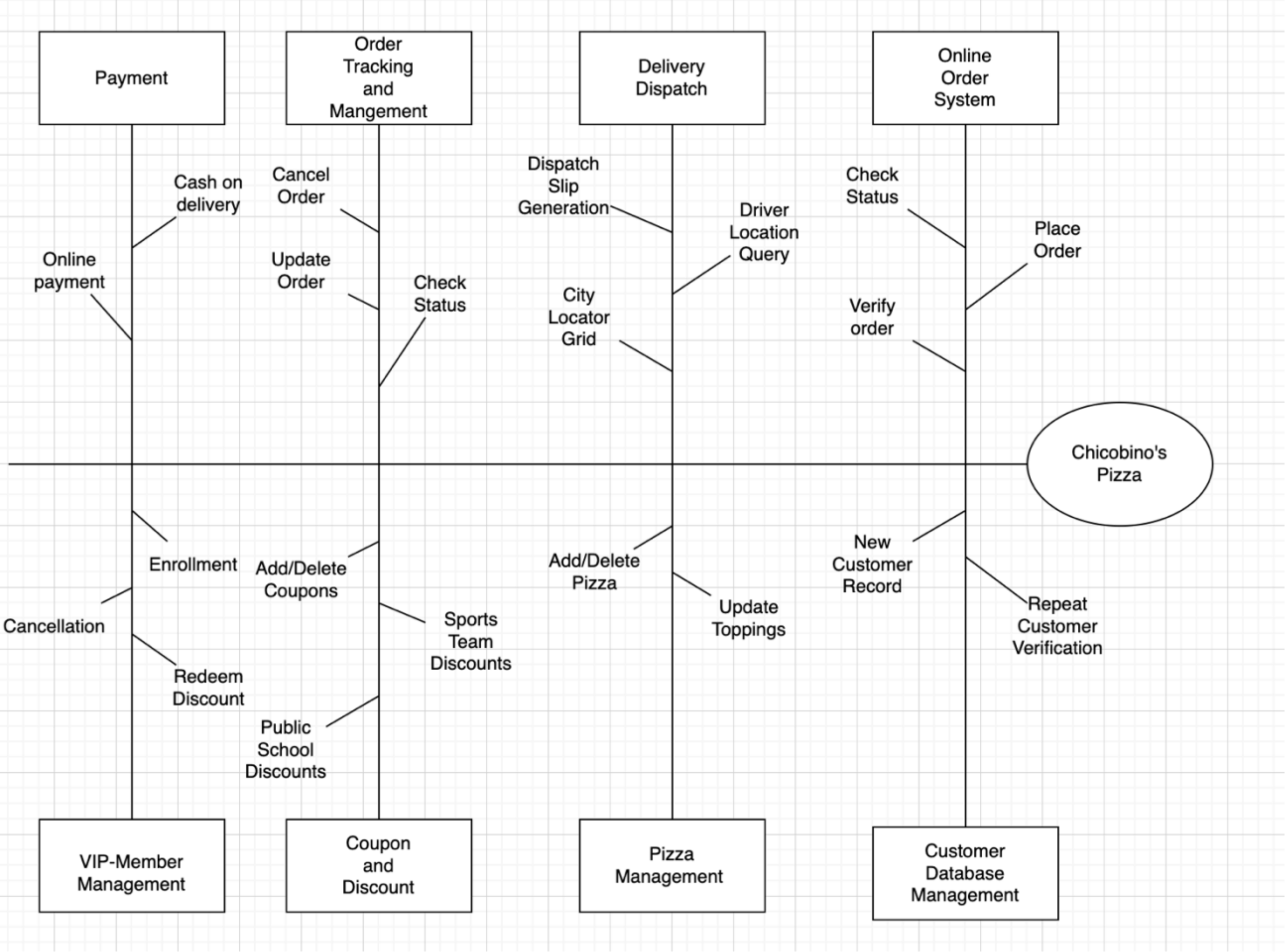
g) Payment System
   i. Customer pays via credit card online.
   ii. Customer opts to pay cash on delivery.
   iii. Salesman checks if cash on delivery is permitted for an address.

h) VIP-Member Management
   i. Customer enrolls as a VIP member.
   ii. Customer cancels VIP membership.
   iii. Customer redeems VIP discount.

4. Context Diagram



5. Feature Tree


Payment
Order Tracking and Mangement
Delivery Dispatch
Online Order System
Cash on delivery
Online payment
Cancel Order
Update Order
Check Status
Dispatch Slip Generation
Driver Location Query
City Locator Grid
Check Status
Place Order
Verify order
Chicobino's Pizza
Enrollment
Cancellation
Redeem Discount
Add/Delete Coupons
Sports Team Discounts
Public School Discounts
Add/Delete Pizza
Update Toppings
New Customer Record
Repeat Customer Verification
VIP-Member Management
Coupon and Discount
Pizza Management
Customer Database Management