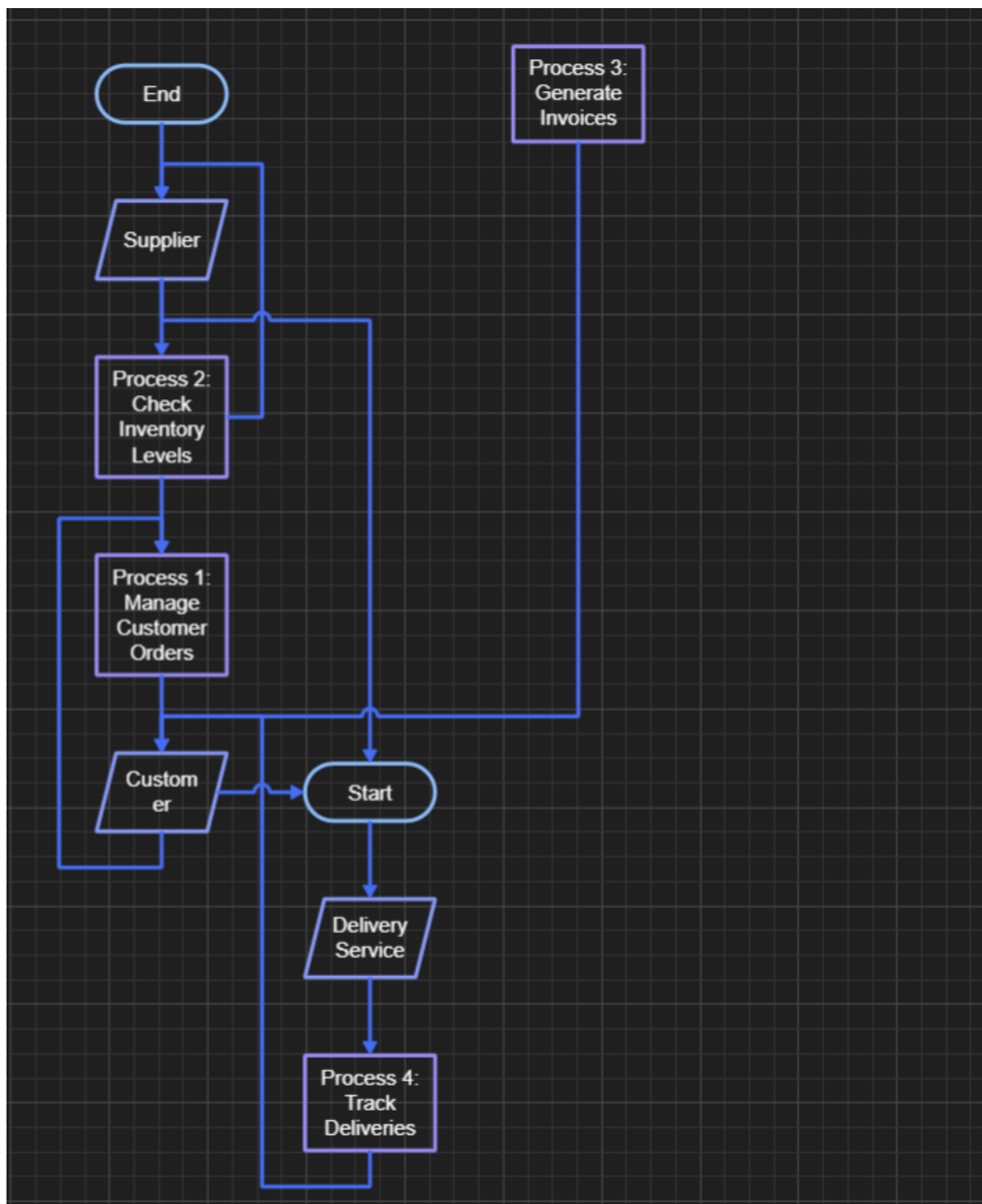


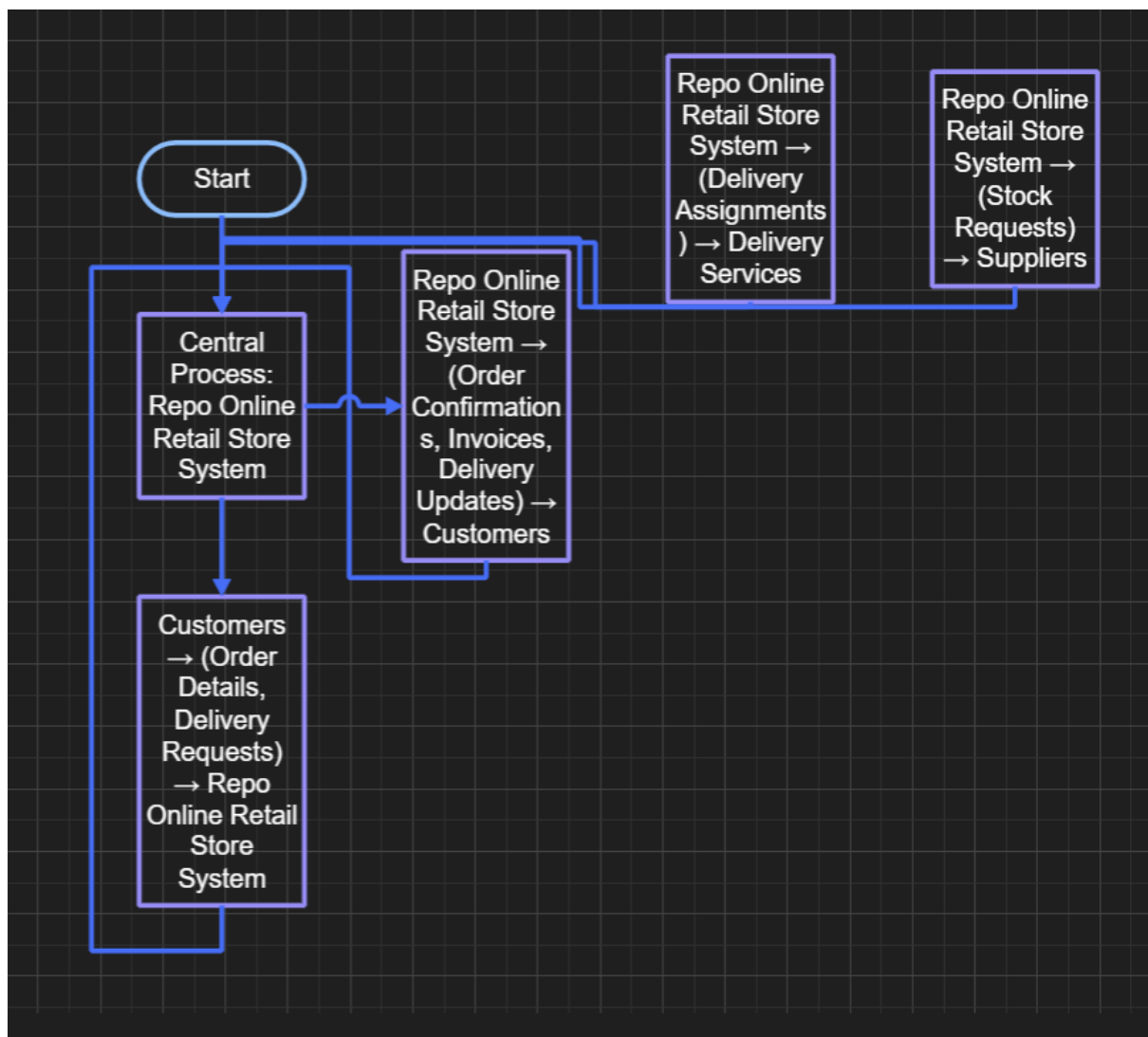
Tabish Parkar

Formative
Assessment 1:
Systems Analysis
And
Design
05HA2309970
Bellville

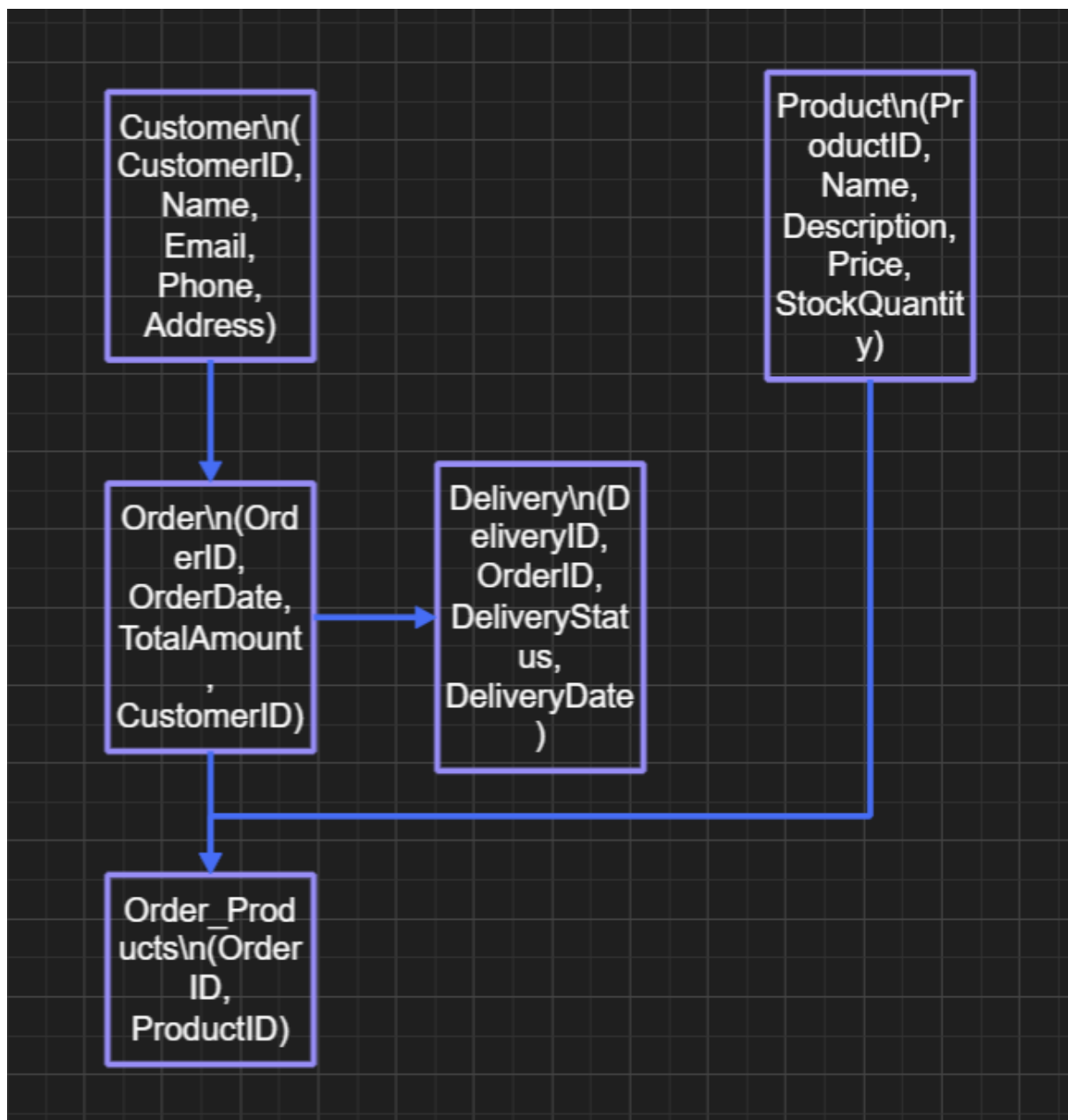
1A



1B.



1C.



1D. Clarity – Data Flow Diagrams, context diagrams and Entity Relationship Diagrams show complex system interactions and structures in a clear and visual format.

Communication - They allow for effective communication between stakeholders, including developers, business analysts and clients.

Requirement Capture - These models allow for aid to be used in capturing and understanding system requirements before any development begins which in turn reduces errors.

Problem Identification - They can find probable issues in system design and interactions young into the development process.

Foundation for Development - These models serve as a blueprint for system design and development which allows for a structured progression of work.

2. A.

Systematic Planning – Fosters a careful approach to planning, which in turn leads to successful project completion.

Risk Mitigation - Tracks and manages risks during the lifecycle.

Quality Assurance - Constant testing and refining to better overall product quality.

Resource Management - Efficient usage of time, human and capital resources.

User Involvement - Emphasizes feedback from stakeholders which ensures the product meets user needs.

2.B.

Feasibility Studies – Assessing the overall viability of the project with regards to technical, operational and financial points. These will determine if the project should go ahead.

Requirement Analysis - Gathering and clarifying needs of stakeholders, which establishes which functions and features the system needs to provide. This verifies that the development team has a proper understanding of the users' expectations.

3.A.

Benefits:

Goal Orientation – Focuses on measurement efforts on reaching precise project goals.

Clarity - Allows a systematic approach to gain metrics which align with the business needs.

Customization - Promotes adaptability of metrics to domain precise goals.

Limitations:

Overemphasis on Goals - Could fall to disregarding various other critical aspects that do not align with the set goals.

Complexity - Fostering a goal question metric can be complicated and time consuming.

Interpretation of Results – Outcomes could be misjudged if the goals are not properly defined.

3.B.

Product Metrics - Assessing the quality and performance of the final product, finding design errors and usage concerns.

Project Metrics - Monitoring project statistics, verifying proper resource allocation and following the deadline.

Process Metrics - Testing development methods, improving team efficiency and minimizing waste.

3.C.

Product metrics helps to provide information or statistics during testing. Metrics also help in the starting and planning stages by allowing for timelines and budgets to be set.

Furthermore, they provide insight into adjustments while execution is taking place, which also supports iteration and helps to optimize performance.

3.D.

Advantages:

Timeliness - Minimized measurement periods allow for timely feedback, which allows for rapid adjustments.

Focus on Trends - Trend analysis emphasizes patterns and advancements, allowing for a practical approach.

Disadvantages:

Short-Term Focus - This might possibly lead to disregarding long-term quality trends.

Increased Pressure - May cause stress on the team to constantly meet evolving metrics.

3.E.

Risks – Exaggerating metrics could lead to a loss in software quality as teams focus on numerical goals disregarding overall user satisfaction.

Mitigation Strategies - Customer satisfaction goals need to be aligned with metrics.

Prioritizing quality feedback along with measurable metrics. Lastly , adopting a culture emphasizing usability and customer satisfaction along with the metrics.