Lab 2.1 What methodology to choose?

A table with text on it

Description automatically generated

Top of Form

1. Suppose that you are an analyst developing a new information system to automate the sales transactions and manage inventory for each retail store in a large chain. The system would be installed at each store and would exchange data with a mainframe computer at the company’s head office. What methodology would you use? Why?

**User requirements:**

* Sales Transaction Management
* Inventory Management
* Data Synchronization with Head Office
* Store-Specific Configuration: Allow for specific product pricing, promotions, and discounts to be configured at each individual store.
* Product Management: Allow store managers to upload new products or update existing ones (e.g., changing prices or stock quantities) in bulk via CSV or another simple interface.
* Sales Forecasting and Inventory Planning
* Promotions and Loyalty Program Integration
* Customer Data and Support

**Familiarity with Technology:**

* Existing Systems: The project should ideally use technologies familiar to both the retail store staff and the IT team at the head office. If stores are already using certain POS systems or inventory management tools, it's beneficial to align with or extend those systems rather than introducing entirely new, unfamiliar solutions.
* Data Exchange with Mainframe: Since the system must exchange data with a mainframe at the head office, ensuring compatibility with the existing mainframe infrastructure is important.

**Complexity:** This project will be quite complex because of the number of different features and legacy systems that will have to be considered. Many of the requirements listed above are interconnected and will have to work coherently with each other, thus increasing the complexity of the relationships between those muddles.

**Time Schedule:**

Assuming the business which the software is developed for isn’t brand new and was functional in the past, it most likely had some system in place. Thus, the store can operate without immediate implementation of this project. Though, the company might be eager to start using the new system and that would require prioritizing the essentials modules of the project first.

Priority List:

1. POS System
2. Product Management
3. Inventory management
4. Data Synchronization with Head Office

**Methodology:**

**Throwaway Prototyping** would fit the best for this project. Retail store needs and integration with the mainframe may require different attempts before a working solution is found. Throwaway prototyping will help to achieve high reliability, which is crucial for such systems, because it allows the developers to make small prototypes before committing to the solution. This methodology allows regular testing during each prototype, ensuring that reliability is maintained throughout the project.

1. Suppose that you are an analyst developing a new executive information system (EIS) intended to provide key strategic information from existing corporate databases to senior executives to help in their decision making. What methodology would you use? Why?

**User Requirements:**

* Key Metrics and Strategic Information: The system must provide senior executives with accurate, high-level insights from existing corporate databases to assist in decision-making.
* Data Integration: Seamless extraction, transformation, and loading processes are essential to consolidate data from multiple corporate sources into the EIS.
* Customizable Dashboards and Reports: Executives should be able to view relevant information in an intuitive, user-friendly interface with customization options for key performance indicators.

**Familiarity with Technology:**

* Existing Databases and Tools: The EIS should leverage existing corporate databases and reporting tools wherever possible. It should be integrated seamlessly with the organization’s current IT infrastructure.
* User-Friendly Interface: Since senior executives may not be highly technical, the system interface must prioritize ease of use with minimal learning curve requirements.
* Report Exporting and Sharing: The system should include options for exporting reports in formats like PDF or Excel and allow secure sharing across teams.

**Complexity:**  
This project is moderately complex due to the need to extract and transform data from various corporate databases while ensuring data integrity and compatibility. The system also requires a high level of customization for dashboards and real-time data updates, making it essential to get the requirements right up front.

**Time Schedule:**

Implementation of the time of the project is important but not crucial because the company existed for quite some time and that has already developed ways to use the data they have. This system is supposed to make it easier for the executives to make decisions, but the business can still function while the system is in development.

**Methodology:**  
**Iterative Development** is most suitable for this project.

**Reasoning:**

* Incremental Delivery: While the metrics, dashboards, and reports are generally well-defined, Iterative Development allows the project to proceed with flexibility, delivering small, functional increments of the system. Each iteration focuses on specific objectives, ensuring continuous progress.
* User-Centered Feedback: Iterative Development emphasizes regular feedback at the end of each iteration. This approach ensures that the system meets the high accuracy and reliability needs of senior executives by refining features based on user input.
* Data Integration Refinement: The iterative approach allows seamless integration and transformation of data from multiple sources to be addressed incrementally. This reduces errors and ensures data integrity by testing ETL processes in smaller, manageable stages.
* Adaptability to Evolving Needs: While executive requirements may appear stable, Iterative Development offers the flexibility to adapt if new insights or needs emerge during the development process. This minimizes resource wastage on rework.
* Continuous Testing: Testing is embedded in every iteration, ensuring accuracy in ETL processes, reliability of dashboards, and correctness of reports. Early and frequent testing reduces the risk of critical issues at later stages.

By focusing on iterative cycles, this methodology ensures the system evolves in line with executive expectations while maintaining high standards of functionality and reliability.

1. Suppose that you are an analyst working for a small company to develop an accounting system. What methodology would you use? Why?

**User Requirements:**

* Core Accounting Features: The system must include standard accounting functions such as accounts payable, accounts receivable, general ledger, payroll, and financial reporting.
* Budget Management: Enable the company to track budgets and compare them with actual expenses.
* Tax Compliance: Ensure the system adheres to local tax regulations and automates tax calculations and reporting.
* Customizable Reports: Provide options for generating customized financial reports tailored to the company's specific needs.
* Ease of Use: Since this is a small company, the system should be intuitive and easy for non-technical users to navigate.

**Familiarity with Technology:**

* Existing Tools: The company may already be using simple tools like spreadsheets or basic accounting software. The new system should integrate with or replace these tools without a steep learning curve.
* Cloud vs. Local Deployment: Depending on the company’s preferences, the system could either be cloud-based for accessibility or locally installed for better control.

**Complexity:**  
The project is moderately complex as it involves designing a robust and scalable system while ensuring simplicity for end users. Integration with existing tools and compliance with tax regulations add layers of complexity.

**Time Schedule:**

The business is an established small company and not a startup in early development, thus they already have some way of doing accounting. So, although the company might require the project to be finished fast there should be some flexibility. The company’s function is not reliant on the system and the project can take as much time as required.

**Methodology:**  
**V-Model** is the most suitable methodology for this project.

**Reasoning:**

* Structured Development and Testing: The V-Model emphasizes a rigorous, structured approach, where each phase of development is paired with corresponding testing. This is particularly beneficial for an accounting system where accuracy and compliance are critical. For example, the payroll feature will undergo validation and verification at every step to ensure they meet the specified requirements.
* Clear Requirements and Design: The V-Model requires detailed upfront planning and design, making it ideal for projects where the requirements are stable and well-understood, such as in an accounting system. Each requirement will be thoroughly defined and then developed and tested in parallel, ensuring that the final system adheres to business needs.
* Emphasis on Compliance and Accuracy: By following the V-Model, the system will undergo comprehensive validation and testing, ensuring that all features (e.g., tax calculations, financial reporting) meet regulatory standards. This approach is crucial for small businesses that must comply with local financial regulations.
* Reduced Risk through Early Testing: Testing occurs early in the process (alongside development), helping to identify potential issues early on and mitigating the risk of large-scale failures. Features like accounts payable and reporting will be verified during the design and development phases to ensure they align with expectations.
* Predictability and Documentation: The V-Model’s focus on clear documentation and sequential phases makes it a great choice for ensuring that the business’s needs are met predictably.

**Conclusion:**

By using the **V-Model**, the accounting system can be developed with a clear structure, thorough testing, and rigorous validation at each stage. This ensures that the system will meet the company’s needs for accuracy, compliance, and functionality while reducing the risk of issues arising later in the development process.