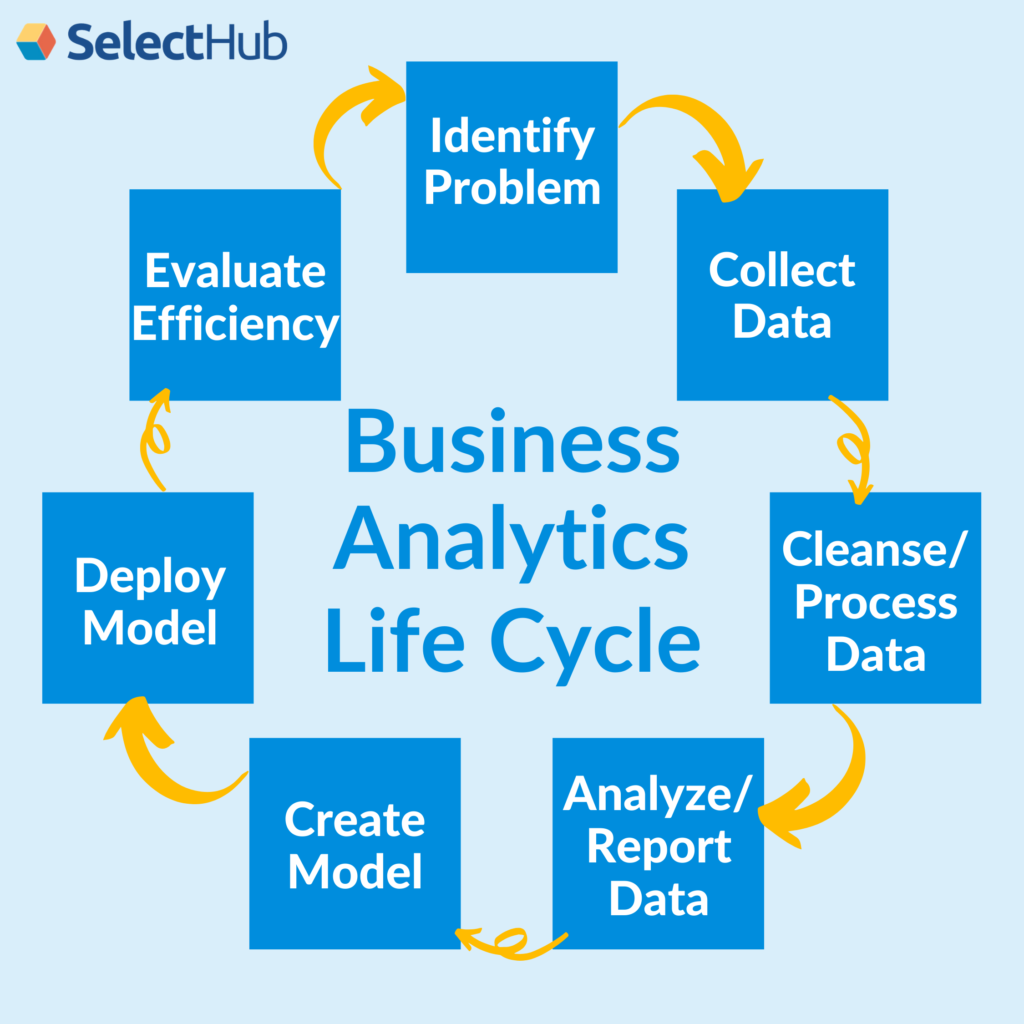
**Name : Sanket Gulhane , DA-14**

* **what is Business Analytics?**

Business Analytics (BA) involves leveraging data, statistical analysis, and modeling techniques to make more informed business decisions. By examining data, organizations can gain valuable insights into their performance and uncover opportunities for improvement. Business analysts, data scientists, and data analysts are typically responsible for carrying out these processes, which aim to optimize decision-making and drive business growth.

One of the key components of Business Analytics is data collection and cleaning. This involves gathering data from various internal and external sources, such as databases, datasets, and market research, and then preparing it for analysis. Data cleaning ensures that the information is accurate and usable, providing a solid foundation for further analysis. Once the data is ready, different types of analytics come into play, including descriptive analytics, which focuses on understanding past events by analyzing historical data to identify trends and patterns. Predictive analytics takes it a step further by using statistical models and machine learning techniques to forecast future outcomes, such as sales trends or customer behavior. Prescriptive analytics, on the other hand, provides actionable recommendations to optimize business performance, often through simulations and optimization algorithms. Additionally, diagnostic analytics helps businesses understand why something happened by exploring the root causes behind specific outcomes.

**Business Analytics life Cycle :**



The roles in Business Analytics can vary depending on the specific area of focus. A **business analyst** is responsible for identifying business problems and providing data-driven solutions, working closely with stakeholders to interpret the data and suggest improvements. A data analyst focuses on data collection, processing, and visualization, using statistical tools to analyze trends and generate reports. With more advanced technical skills, a data scientist works on large datasets and employs machine learning and predictive modeling techniques to derive insights. Business Intelligence (BI) analysts specialize in using analytics tools to create dashboards and reports that track business performance and provide actionable insights. Finally, data engineers design and build the infrastructure to collect, store, and analyze data, ensuring smooth data flow across systems.

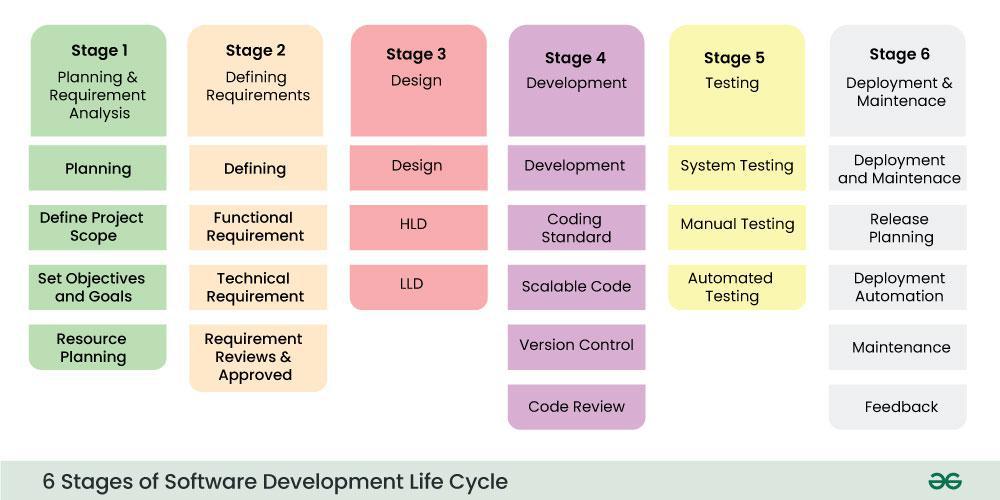
Business Analytics professionals engage in a range of activities. They conduct dataanalysis on both structured and unstructured data to uncover insights that inform decision-making. They are also responsible for reporting, which involves generating clear and concise dashboards, reports, and visualizations to present complex data in an accessible way for non-technical stakeholders. Forecasting is another key responsibility, where historical data is used to predict future trends and behaviors, guiding business strategies. Business analytics professionals also provide decisionsupport, offering actionable recommendations to business leaders based on their findings. Moreover, they work on optimization, identifying inefficiencies in processes and suggesting improvements to reduce costs and increase efficiency. Lastly, they often focus on automation, developing systems to streamline repetitive tasks and improve business operations' overall accuracy and efficiency.

**Tools and Technologies:**

Business analytics professionals use a variety of tools and technologies, including:

* Statistical Analysis Tools: R, Python, SPSS
* Data Visualization: Tableau, Power BI, QlikView
* Databases: SQL, NoSQL databases like MongoDB, Hadoop for big data
* Business Intelligence Tools: SAS, SAP BusinessObjects
* Predictive Modeling & Machine Learning: Python libraries (Scikit-learn, TensorFlow, etc.), Azure Machine Learning, IBM Watson
* **What is software Development Life Cycle SDLC :**

Software development life cycle (SDLC) is a structured process that is used to design, develop, and test good-quality software. SDLC, or software development life cycle, is a methodology that defines the entire procedure of software development step-by-step. The goal of the SDLC life cycle model is to deliver high-quality, maintainable software that meets the user’s requirements. SDLC in software engineering models outlines the plan for each stage so that each stage of the software development model can perform its task efficiently to deliver the software at a low cost within a given time frame that meets users requirements. In this article we will see Software Development Life Cycle (SDLC) in detail.



* Stages Of Software Development Life Cycle

**Stage-1: Requirement Analysis :**

Planning is a crucial step in everything, just as in Software Development In this same stage, [Requirement analysics](https://www.geeksforgeeks.org/activities-involved-in-software-requirement-analysis) is also performed by the developers of the organization. This is attained from customer inputs, and sales department/market surveys.

The information from this analysis forms the building blocks of a basic project. The quality of the project is a result of planning. Thus, in this stage, the basic project is designed with all the available information.

**Stage-2: Defining Requirements :**

In this stage, all the requirements for the target software are specified. These requirements get approval from customers, market analysts, and stakeholders.   
This is fulfilled by utilizing SRS (Software Requirement Specification). This is a sort of document that specifies all those things that need to be defined and created during the entire project cycle.

**Stage-3: Designing Architecture :**

[SRS](https://www.geeksforgeeks.org/software-requirement-specification-srs-format) is a reference for software designers to come up with the best architecture for the software. Hence, with the requirements defined in SRS, multiple designs for the product architecture are present in the Design Document Specification (DDS).

This DDS is assessed by market analysts and stakeholders. After evaluating all the possible factors, the most practical and logical design is chosen for development.

**Stage-4: Developing Product :**

At this stage, the fundamental development of the product starts. For this, developers use a specific programming code as per the design in the DDS. Hence, it is important for the coders to follow the protocols set by the association. Conventional programming tools like compilers, interpreters, debuggers, etc. are also put into use at this stage. Some popular languages like C/C++, Python, Java, etc. are put into use as per the software regulations.

**Stage-5: Product Testing and Integration :**

After the development of the product, testing of the software is necessary to ensure its smooth execution. Although, minimal testing is conducted at every stage of SDLC. Therefore, at this stage, all the probable flaws are tracked, fixed, and retested. This ensures that the product confronts the quality requirements of SRS.

**Stage-6: Deployment and Maintenance of Products :**

After detailed testing, the conclusive product is released in phases as per the organization’s strategy. Then it is tested in a real industrial environment. It is important to ensure its smooth performance. If it performs well, the organization sends out the product as a whole. After retrieving beneficial feedback, the company releases it as it is or with auxiliary improvements to make it further helpful for the customers. However, this alone is not enough. Therefore, along with the deployment, the [product’s supervision](https://www.geeksforgeeks.org/product-management).

* **Agile Methodology Sprint in Agile :**

In Agile methodology, a **Sprint** is a time-boxed iteration or cycle in which a team works to complete a set of defined tasks, typically referred to as **user stories**, from the project backlog. Sprints are a core component of Agile frameworks like Scrum and are designed to deliver incremental value through the development of a working product or feature.

A Sprint typically lasts between **1 and 4 weeks**, with **2 weeks** being the most common duration. The duration is fixed and remains consistent throughout the project to ensure regular, predictable delivery. At the beginning of each Sprint, the team conducts a **Sprint Planning** meeting. During this meeting, the **Product Owner** presents the highest-priority items from the product backlog. The **Development Team** discusses and selects which items (user stories, tasks, or features) they can realistically complete during the Sprint, considering their capacity. Additionally, the team defines a **Sprint Goal**, which is a clear, concise objective that provides focus and aligns the team’s efforts.

During the Sprint, the team works on the selected items, often in daily **stand-up meetings** (also called **Daily Scrums**) to assess progress, address any roadblocks, and adjust plans if necessary. The team is responsible for managing its own workflow, and members collaborate to complete the tasks. They may break down the work into smaller tasks, use tools like **Kanban boards** or **task boards**, and apply other Agile practices like **pair programming** or **test-driven development**.



After the Sprint Review, the team holds a **Sprint Retrospective** to reflect on the process and the team’s performance. The team discusses what went well, what didn’t go well, and what improvements can be made in the next Sprint. This reflection is essential for continuous improvement and ensuring that the team evolves in terms of both process and collaboration.

The outcome of a Sprint is typically a **potentially shippable product increment**—a piece of the product that is fully developed, tested, and ready for delivery or release (depending on the project's release strategy). Not every Sprint will result in a fully "shippable" product, but the goal is to have a functional and tested increment at the end of each Sprint.

A key principle of a Sprint is that it is **time-boxed**, meaning the duration is fixed, and no work should extend beyond the Sprint's time frame. Teams are also expected to make a **commitment to the Sprint Goal**, ensuring that the agreed-upon work is delivered in alignment with the Sprint’s objectives. **Transparency** is another key element, meaning that the progress, issues, and work done during the Sprint should be visible to all stakeholders. While teams are encouraged to be **adaptable**, any changes or adjustments to the work should occur at the beginning of the next Sprint, or during the Sprint Review if necessary. **Collaboration** is at the heart of the process, with team members working closely together to achieve the Sprint Goal.

Sprints offer numerous benefits. They enable **frequent delivery of value** by ensuring that the team delivers working software at the end of every Sprint, which provides regular value to customers or stakeholders. The **continuous improvement** inherent in the Sprint Retrospective allows teams to refine their processes over time. Sprints also provide **predictability** since each Sprint has a fixed duration and defined deliverables, making it easier to plan and track progress. Furthermore, the iterative approach allows for **flexibility**, as priorities and the product itself can be adjusted based on feedback after each Sprint.

**Benefits of Sprints:**

* **Frequent Delivery of Value**: By delivering working software at the end of every Sprint, teams can provide frequent value to customers or stakeholders.
* **Continuous Improvement**: Sprints allow for regular reflection, making it easier to improve the team’s processes over time.
* **Predictability**: Sprints allow teams to plan and predict progress, as each Sprint has a defined duration and set of deliverables.
* **Flexibility**: Teams can adjust priorities and refine the product based on feedback after each Sprint, making it easier to respond to changes.
* **Sprint Wise 4 Month Health Care Website Project Plan**

1. **Sprint 1: Planning and UI/UX Design (Weeks 1–2) :**

The team focuses on gathering requirements and creating the project roadmap. The UI/UX Designer develops wireframes and user flow diagrams to visualize the website’s interface, while the Compliance Specialist identifies regulatory requirements for handling healthcare data.

Deliverables: Project plan, approved wireframes, and compliance checklist.

Teams: Project Manager, UI/UX Designer, Compliance Specialist.

1. **Sprint 2: Back-End Architecture Design (Weeks 3–4) :**

The Back-End Developer designs the database schema and plans the server-side architecture. Security and compliance are emphasized in this stage, ensuring the platform supports healthcare data standards.

Deliverables: Database schema and back-end design blueprint.

Teams: Back-End Developer, Compliance Specialist.

1. **Sprint 3: Data Collection and Preprocessing (Weeks 5–6) :**

The Data Scientist and ML Engineer collect healthcare datasets and preprocess the data to ensure it is clean, complete, and ready for machine learning. This involves handling missing data, normalization, and feature engineering.

Deliverables: Preprocessed dataset ready for ML training.

Teams: Data Scientist, ML Engineer.

1. **Sprint 4: Front-End Development (Phase 1) (Weeks 7–8) :**

The Front-End Developer creates the basic structure of the website, including the landing page, login system, and user registration. The UI/UX Designer ensures the design aligns with user-friendly principles.

Deliverables: Functional front-end for core pages.

Teams: Front-End Developer, UI/UX Designer.

1. **Sprint 5: ML Model Training and Evaluation (Weeks 9–10) :**

Machine learning models are trained using the preprocessed data. The ML Engineer focuses on algorithms for features like symptom prediction and risk analysis, while the Data Scientist evaluates the models for accuracy and reliability.

Deliverables: Trained ML models with performance metrics.

Teams: ML Engineer, Data Scientist.

1. **Sprint 6: Integration of Front-End, Back-End, and ML Models (Weeks 11–12) :**

The system’s components are integrated to ensure smooth functionality. The Front-End Developer connects the user interface with the back-end APIs, and the ML models are deployed to provide real-time results.

Deliverables: Fully functional and integrated system.

Teams: Front-End Developer, Back-End Developer, ML Engineer

1. **Sprint 7: QA Testing and Compliance Validation (Weeks 13–14) :**

The QA Tester identifies and resolves bugs while testing the website’s performance and security. The Compliance Specialist ensures that the platform adheres to regulations like HIPAA or GDPR.

Deliverables: Bug-free, compliant website.

Teams: QA Tester, Compliance Specialist.

1. **Sprint 8: Deployment and Launch (Weeks 15–16) :**

The DevOps Engineer deploys the website to a cloud platform and establishes CI/CD pipelines for continuous updates. The team monitors the website post-launch, collects user feedback, and resolves critical issues.

Deliverables: Live healthcare website with deployment pipeline.

Teams: DevOps Engineer, Entire Team.

**Conclusion:** Each sprint targets specific milestones, progressing systematically from planning to deployment. By Sprint 8, the healthcare website is live, secure, and ready for users, incorporating ML-powered features and compliance with healthcare standards.