

A collage of business documents, charts, and hands working on a project. The background is a dark, textured surface. Overlaid on this are various white documents and papers. Some papers show line graphs, bar charts, and pie charts. One document has a map of the world. Another document has a line graph with a red circle around a peak. A hand is pointing at a document with a red pen. Another hand is holding a blue pen. A third hand is holding a yellow pen. A fourth hand is holding a green pen. A fifth hand is holding a pink pen. A sixth hand is holding a blue pen. A seventh hand is holding a yellow pen. A eighth hand is holding a green pen. A ninth hand is holding a pink pen. A tenth hand is holding a blue pen. A eleventh hand is holding a yellow pen. A twelfth hand is holding a green pen. A thirteenth hand is holding a pink pen. A fourteenth hand is holding a blue pen. A fifteenth hand is holding a yellow pen. A sixteenth hand is holding a green pen. A seventeenth hand is holding a pink pen. A eighteenth hand is holding a blue pen. A nineteenth hand is holding a yellow pen. A twentieth hand is holding a green pen. A twenty-first hand is holding a pink pen. A twenty-second hand is holding a blue pen. A twenty-third hand is holding a yellow pen. A twenty-fourth hand is holding a green pen. A twenty-fifth hand is holding a pink pen. A twenty-sixth hand is holding a blue pen. A twenty-seventh hand is holding a yellow pen. A twenty-eighth hand is holding a green pen. A twenty-ninth hand is holding a pink pen. A thirtieth hand is holding a blue pen. A thirty-first hand is holding a yellow pen. A thirty-second hand is holding a green pen. A thirty-third hand is holding a pink pen. A thirty-fourth hand is holding a blue pen. A thirty-fifth hand is holding a yellow pen. A thirty-sixth hand is holding a green pen. A thirty-seventh hand is holding a pink pen. A thirty-eighth hand is holding a blue pen. A thirty-ninth hand is holding a yellow pen. A fortieth hand is holding a green pen. A forty-first hand is holding a pink pen. A forty-second hand is holding a blue pen. A forty-third hand is holding a yellow pen. A forty-fourth hand is holding a green pen. A forty-fifth hand is holding a pink pen. A forty-sixth hand is holding a blue pen. A forty-seventh hand is holding a yellow pen. A forty-eighth hand is holding a green pen. A forty-ninth hand is holding a pink pen. A fiftieth hand is holding a blue pen. A fifty-first hand is holding a yellow pen. A fifty-second hand is holding a green pen. A fifty-third hand is holding a pink pen. A fifty-fourth hand is holding a blue pen. A fifty-fifth hand is holding a yellow pen. A fifty-sixth hand is holding a green pen. A fifty-seventh hand is holding a pink pen. A fifty-eighth hand is holding a blue pen. A fifty-ninth hand is holding a yellow pen. A sixtieth hand is holding a green pen. A sixty-first hand is holding a pink pen. A sixty-second hand is holding a blue pen. A sixty-third hand is holding a yellow pen. A sixty-fourth hand is holding a green pen. A sixty-fifth hand is holding a pink pen. A sixty-sixth hand is holding a blue pen. A sixty-seventh hand is holding a yellow pen. A sixty-eighth hand is holding a green pen. A sixty-ninth hand is holding a pink pen. A seventieth hand is holding a blue pen. A seventy-first hand is holding a yellow pen. A seventy-second hand is holding a green pen. A seventy-third hand is holding a pink pen. A seventy-fourth hand is holding a blue pen. A seventy-fifth hand is holding a yellow pen. A seventy-sixth hand is holding a green pen. A seventy-seventh hand is holding a pink pen. A seventy-eighth hand is holding a blue pen. A seventy-ninth hand is holding a yellow pen. A eightieth hand is holding a green pen. A eighty-first hand is holding a pink pen. A eighty-second hand is holding a blue pen. A eighty-third hand is holding a yellow pen. A eighty-fourth hand is holding a green pen. A eighty-fifth hand is holding a pink pen. A eighty-sixth hand is holding a blue pen. A eighty-seventh hand is holding a yellow pen. A eighty-eighth hand is holding a green pen. A eighty-ninth hand is holding a pink pen. A ninetieth hand is holding a blue pen. A ninety-first hand is holding a yellow pen. A ninety-second hand is holding a green pen. A ninety-third hand is holding a pink pen. A ninety-fourth hand is holding a blue pen. A ninety-fifth hand is holding a yellow pen. A ninety-sixth hand is holding a green pen. A ninety-seventh hand is holding a pink pen. A ninety-eighth hand is holding a blue pen. A ninety-ninth hand is holding a yellow pen. A hundredth hand is holding a green pen.

System Analysis & Design

BUILDING THE FOUNDATION FOR SUCCESSFUL
SYSTEMS

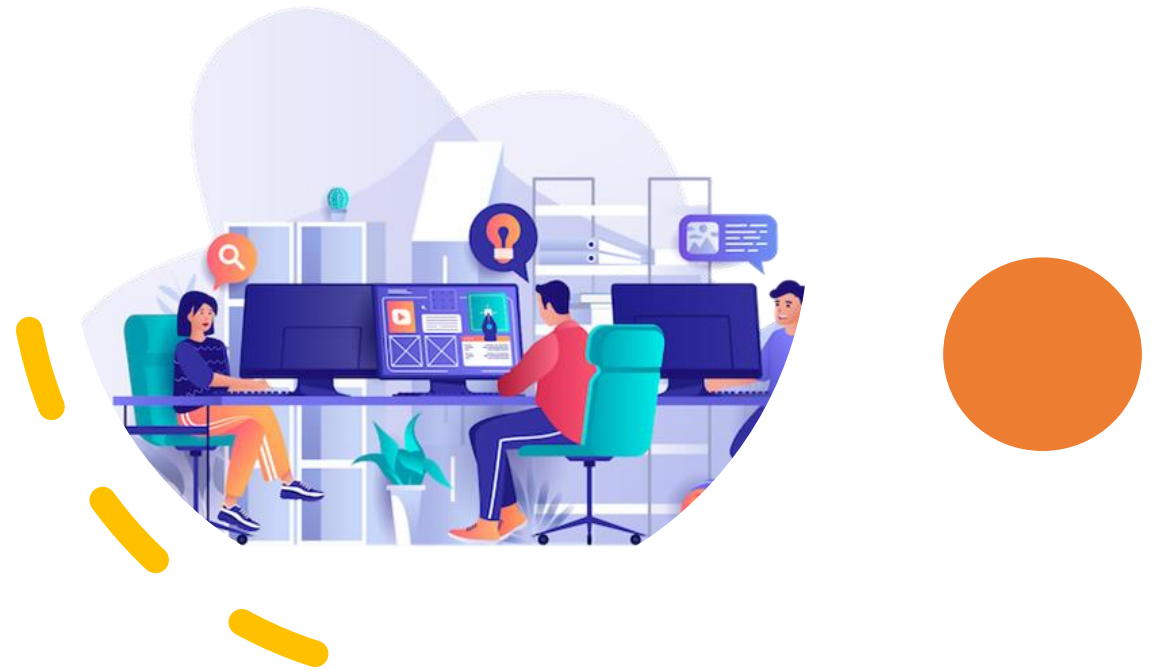
WHAT IS SYSTEM ANALYSIS & DESIGN (SAD)?

- A systematic approach to examining, designing, and implementing efficient and effective systems.
- It involves understanding user requirements, modeling system components, and creating a well-designed architecture.
- SAD ensures that the developed system aligns with organizational goals, meets user needs, and performs optimally.



SYSTEM ANALYSIS & SYSTEM DESIGN

- System Analysis: Process of studying and understanding the current system and its problems, and defining requirements for a new system.



- System Design: Creating a blueprint for the new system, specifying its structure, components, interfaces, and interactions.

WHY SAD IS IMPORTANT?

- Foundation for Success: SAD lays the groundwork for the entire system development process, reducing risks and ensuring a well-designed solution.
- Requirement Understanding: It helps in identifying and documenting user needs, business processes, and system functionalities.
- Efficiency & Effectiveness: SAD optimizes system performance, usability, and scalability while minimizing errors and redundancy.
- Cost & Time Savings: Proper analysis and design lead to fewer reworks, reduced maintenance costs, and faster development cycles.
- Foundation for Success: SAD lays the groundwork for the entire system development process, reducing risks and ensuring a well-designed solution.





BENEFITS OF SYSTEM ANALYSIS & DESIGN

- **Improved Communication:** Clear documentation and visual representations facilitate effective communication between stakeholders, developers, and users.
- **Risk Mitigation:** Identifying potential risks and challenges early in the process helps in proactively addressing them.
- **User-Centric Approach:** SAD ensures that the system meets user expectations, resulting in higher user satisfaction and adoption rates.
- **Scalability & Adaptability:** A well-designed system can accommodate future changes, growth, and technological advancements.

SAD LEADS TO SUCCESSFUL SYSTEMS

- Proper SAD leads to System Success. It ensures:
- Comprehensive Understanding: SAD helps in capturing and analyzing the intricate requirements of a large-scale system, reducing the chances of misunderstandings and misinterpretations.
- Minimizing Errors: Thorough analysis and design processes decrease the likelihood of errors, bugs, and system failures, ensuring a more stable and reliable system.



How SAD Improves Efficiency

- Optimal Resource Utilization: SAD assists in identifying the required system resources, such as hardware, software, and infrastructure, ensuring efficient resource allocation and utilization.
- Streamlining Processes: Analyzing business processes allows for identifying bottlenecks, redundancies, and areas for optimization, leading to streamlined and efficient workflows.





How SAD Facilitates Scalability

- Handling Growing User Base: Large-scale systems typically experience increased user demand over time. SAD helps in designing scalable architectures and components that can handle a growing user base without compromising performance.
- Adapting to Changing Needs: Large-scale systems often require updates, enhancements, and integrations. SAD enables the system to accommodate changes by designing flexible and modular components.

Why is System Analysis & Design Important for a Large-Scale System?

- Large-scale systems have complex requirements, extensive user bases, their interactions and critical dependencies.
- System Analysis & Design (SAD) becomes even more crucial in such scenarios.
- SAD provides the necessary framework to analyze, model, and design complex systems, making them more manageable and understandable.
- By investing time and effort in SAD, organizations can mitigate risks, deliver robust systems, and meet the evolving needs of their users and stakeholders.



System Analysis Process

- Identifying Users and Their Primary Use Cases
- Non-functional Requirements
- Designing Business Processes
- User Experience (UX) Design
- Domain Modeling V1.0



System Design Process

- Domain Modeling V2.0
- Architecture Design
- User Interface (UI) Design
- API Schema Designing
- Database Design
- System Component Design
- Security Design
- Integration Design
- Deployment Plan and Design
- Documentation





Identifying Users and Their Primary Use Cases

- System Analysis

- Identify the different user roles in the system.
- Understand the primary use cases and goals for each user role.
- Analyze the user interactions and workflows to determine the system requirements from a user perspective.

Non-functional Requirements

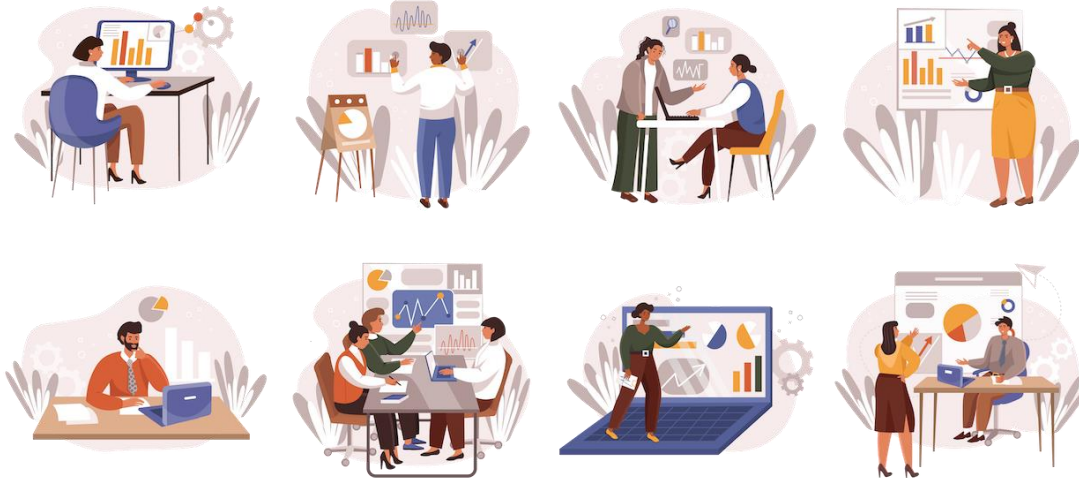
- System Analysis

- Identify and define the non-functional requirements for the job portal system, such as:
 - Performance, security, scalability, and usability.
- Determine the specific metrics, constraints, and guidelines for each non-functional requirement.
- Consider factors like response times, concurrent user capacity, data privacy, and system availability.



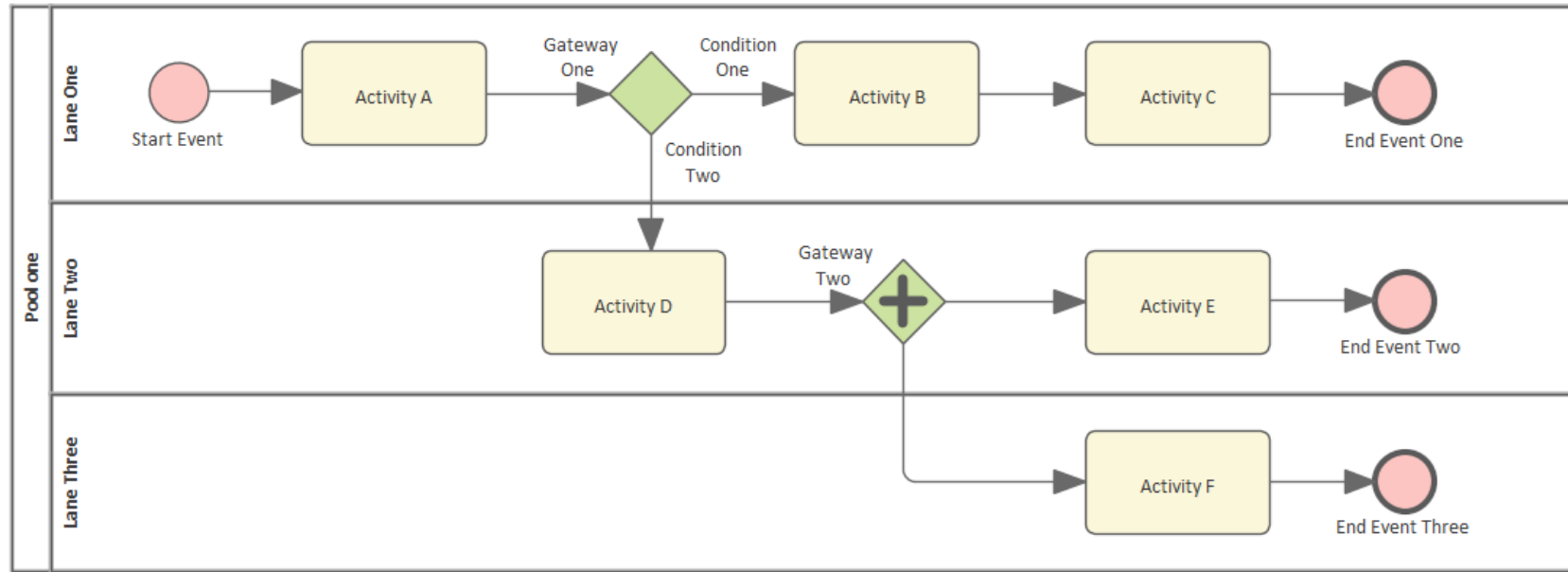
Designing Business Processes

- System Analysis



- Model the key business processes of the system using techniques like BPMN (Business Process Model and Notation).
- Identify the steps, activities, decision points, and data flows within each business process.
- Incorporate error handling and exception scenarios to ensure robust and reliable process flows.

Business Process Diagram with Lanes



Designing Business Processes

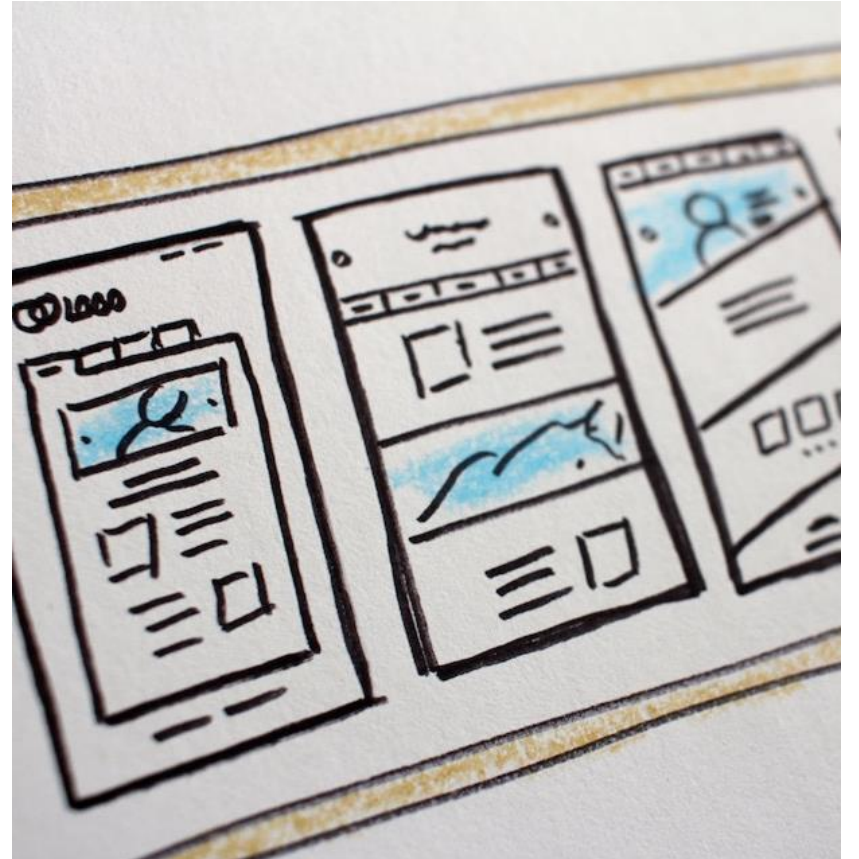
- System Analysis

- Model the key business processes of the system using techniques like BPMN (Business Process Model and Notation).
- Identify the steps, activities, decision points, and data flows within each business process.
- Incorporate error handling and exception scenarios to ensure robust and reliable process flows.

User Experience (UX) Design

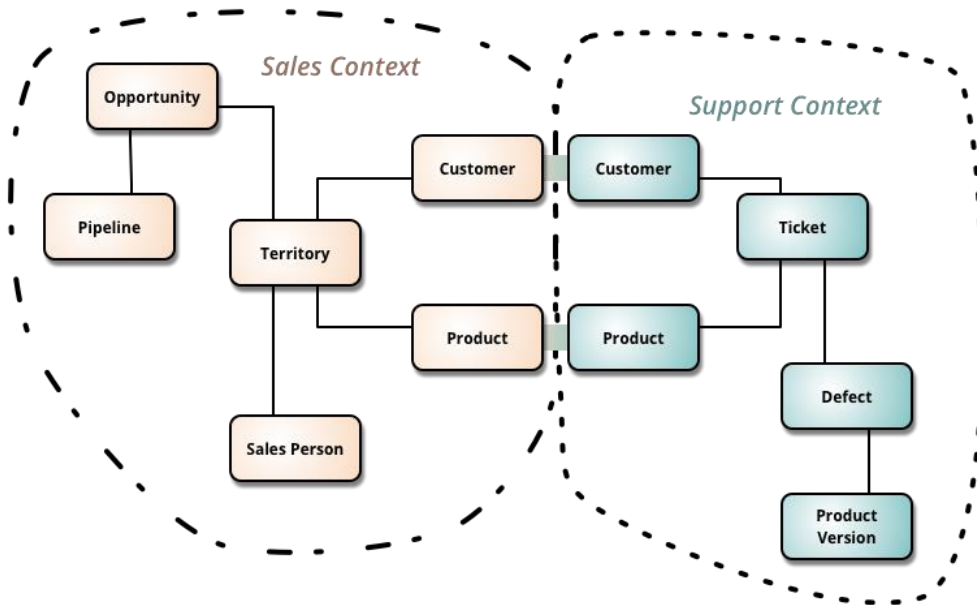
- System Analysis

- Conduct user research and usability analysis to gain insights into user needs and preferences.
- Define user personas and scenarios to understand user behavior and expectations.
- Identify opportunities for enhancing the user experience through effective UI/UX design.



Domain Modeling V1.0

- System Analysis

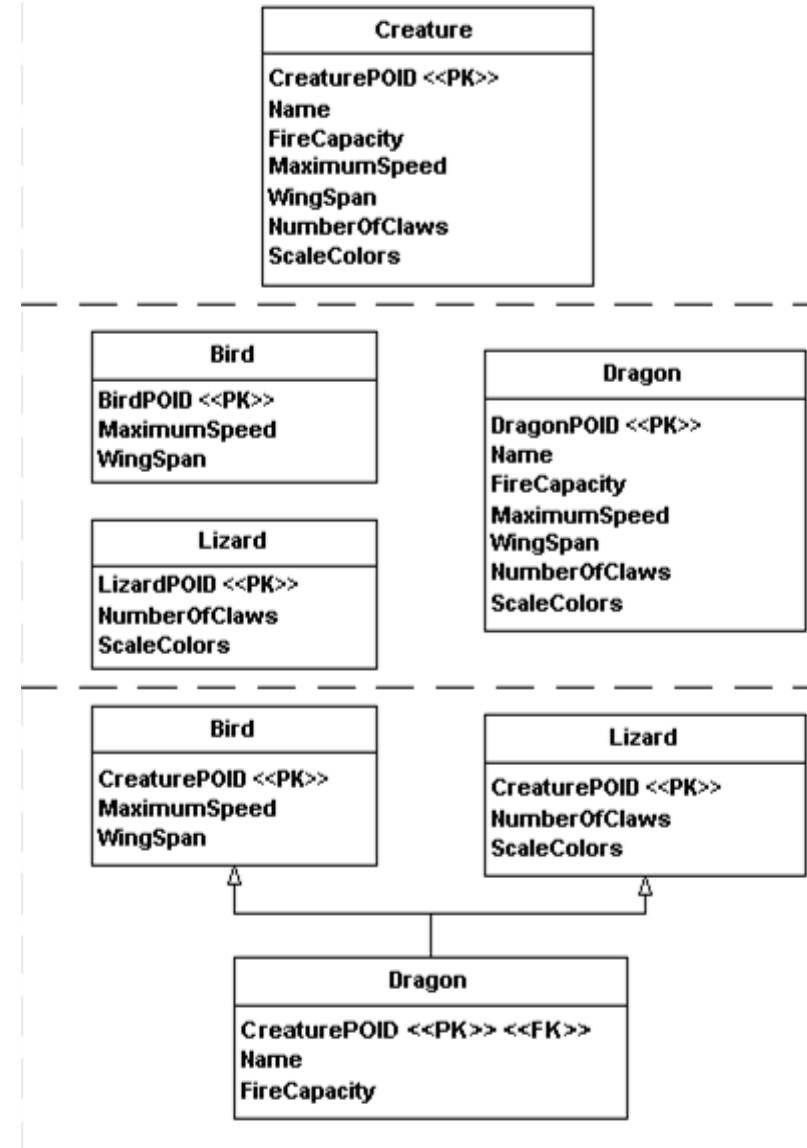


- Perform Conceptual Domain Modeling to identify the key entities, relationships, and behaviors within the job portal system's domain.
- Create a conceptual domain model to capture the high-level structure and dynamics of the domain.
- Perform Logical Domain Modeling to further refine the domain model, incorporating more specific attributes and associations.

Domain Modeling V2.0

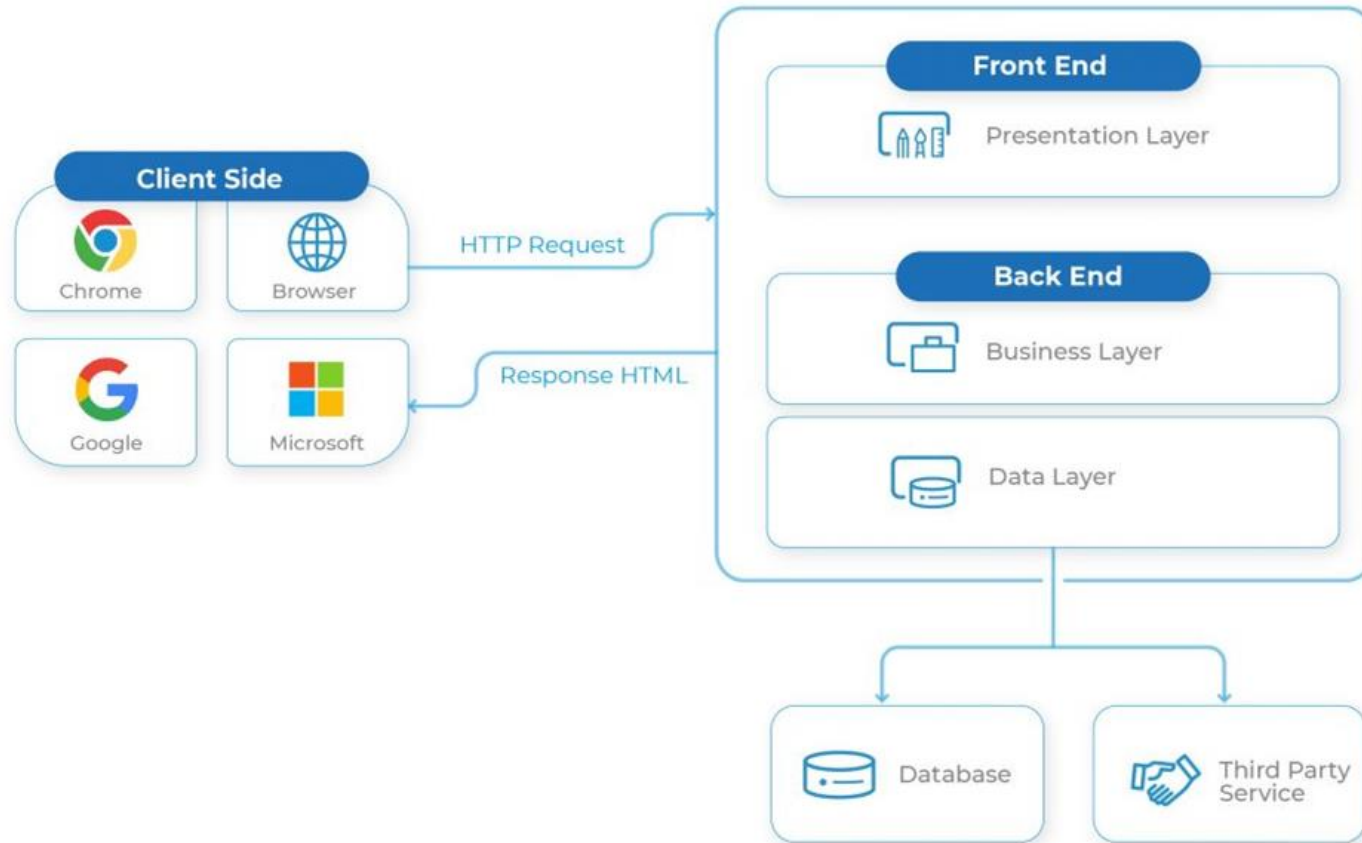
- System Design

- Perform Physical Domain Modeling to map the logical domain model to the database schema design.
- Determine the tables, columns, relationships, and constraints based on the logical domain model.
- Use the physical domain model as the primary input for the Database Schema Design.



Architecture Design

- System Design



- Define the overall system architecture, including the client-side (front-end) and server-side (back-end) components.
- Identify the layers, modules, and subsystems that make up the architecture.
- Determine the communication protocols and interfaces between different components.

User Interface (UI) Design

- System Design

- Incorporate the insights gained from the UX activity performed in the System Analysis stage.
- Design the UI components and layouts to create an intuitive and engaging user experience.
- Consider usability principles, accessibility guidelines, and visual design aesthetics.



The image shows the Swagger Editor interface within the Confluence application. The left pane displays the Swagger JSON schema for a 'pet' API. The right pane shows a visual representation of the API endpoints with their methods and descriptions.

```
1 swagger: "2.0"
2 info:
3   description: "This is a sample server Petstore server. You can find
4     out more about Swagger at [http://swagger.io](http://swagger.io)
5     or on [irc.freenode.net, #swagger](http://swagger.io/irc/).
6     For this sample, you can use the api key `special-key` to test the
7     authorization filters."
8   version: "1.0.0"
9   title: "Swagger Petstore"
10  termsOfService: "http://swagger.io/terms/"
11  contact:
12    email: "apiteam@swagger.io"
13  license:
14    name: "Apache 2.0"
15    url: "http://www.apache.org/licenses/LICENSE-2.0.html"
16 host: "petstore.swagger.io"
17 basePath: "/v2"
18 tags:
19   - name: "pet"
20     description: "Everything about your Pets"
21   - name: "store"
22     description: "Access to Petstore orders"
23   - name: "user"
24     description: "Operations about user"
25 externalDocs:
26   description: "Find out more about our store"
27   url: "http://swagger.io"
28 schemes:
29   - "http"
```

pet Everything about your Pets Find out more: <http://swagger.io>

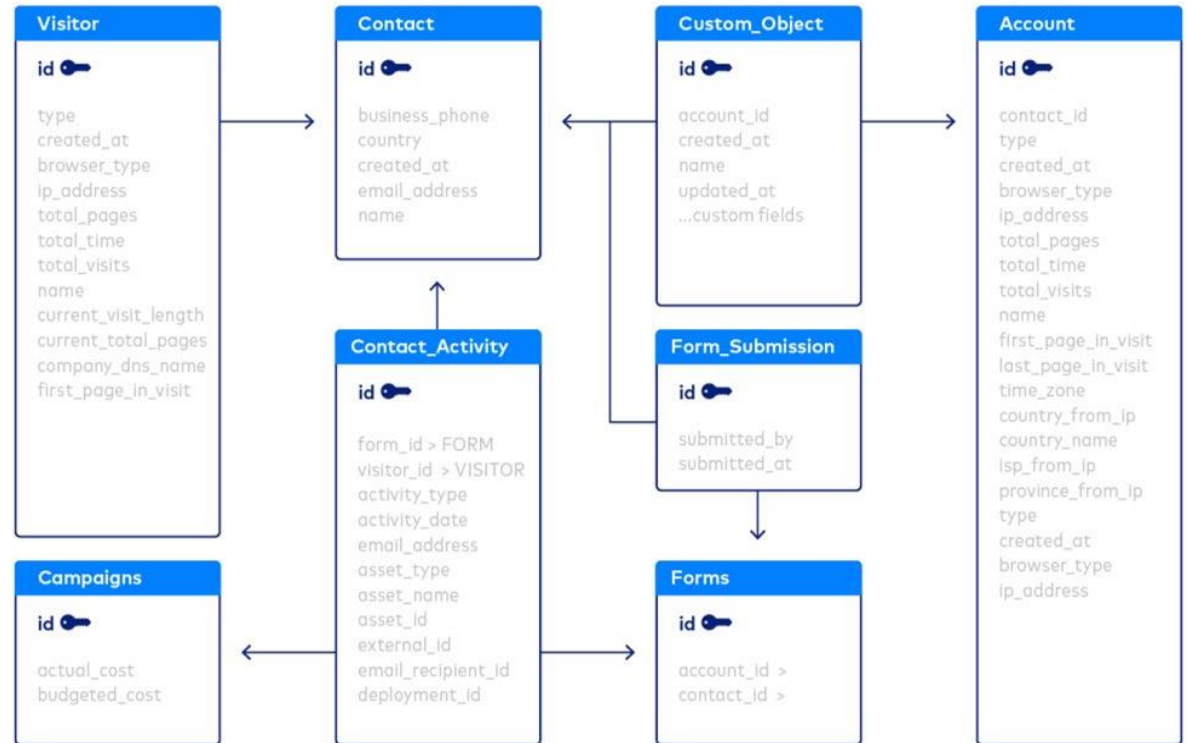
- POST /pet Add a new pet to the store
- PUT /pet Update an existing pet
- GET /pet/findByStatus Finds Pets by status
- GET /pet/findByTags Finds Pets by tags
- GET /pet/{petId} Find pet by ID
- POST /pet/{petId} Updates a pet in the store with form data
- DELETE /pet/{petId} Deletes a pet
- POST /pet/{petId}/uploadImage uploads an image

API Schema Designing - System Design

- Utilize the Logical Domain Model created during the System Analysis phase as an input for API Schema Designing.
- Define the API schema using OpenAPI and Swagger tools such as Swagger Editor.
- Specify the endpoints, request/response structures, and data formats for the job portal system's API.

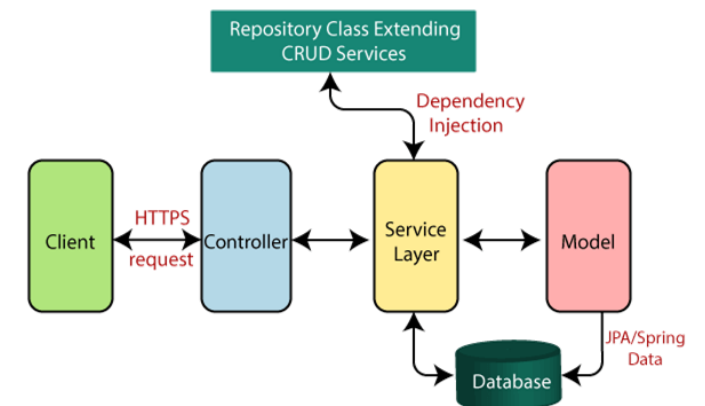
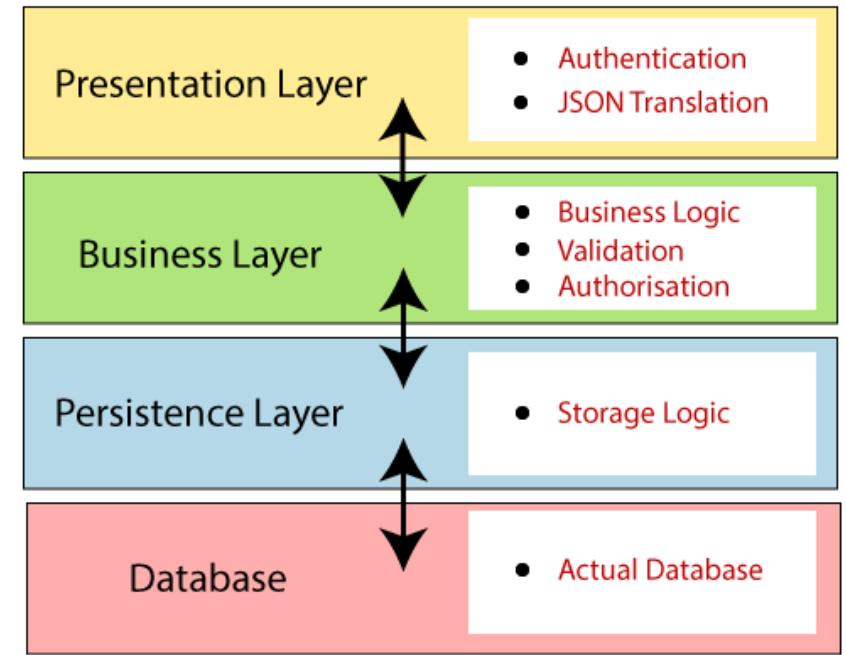
Database Design -System Design

- Utilize the Physical Domain Model created during the System Design phase as a basis for the database schema design.
- Determine the tables, columns, relationships, and constraints based on the physical domain model.
- Optimize the database design for efficient storage, retrieval, and data integrity.

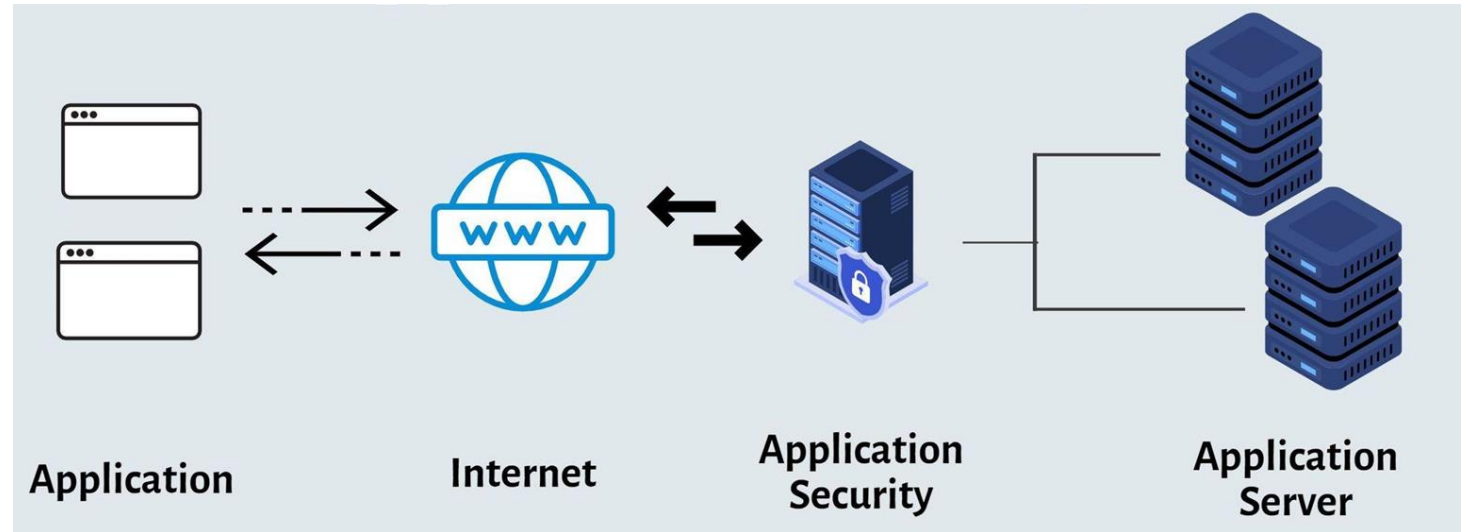


System Component Design -System Design

- Break down the system architecture into smaller components and modules.
- Define the functionalities, responsibilities, and interactions of each component.
- Determine the technologies, frameworks, and libraries to be used for each component.

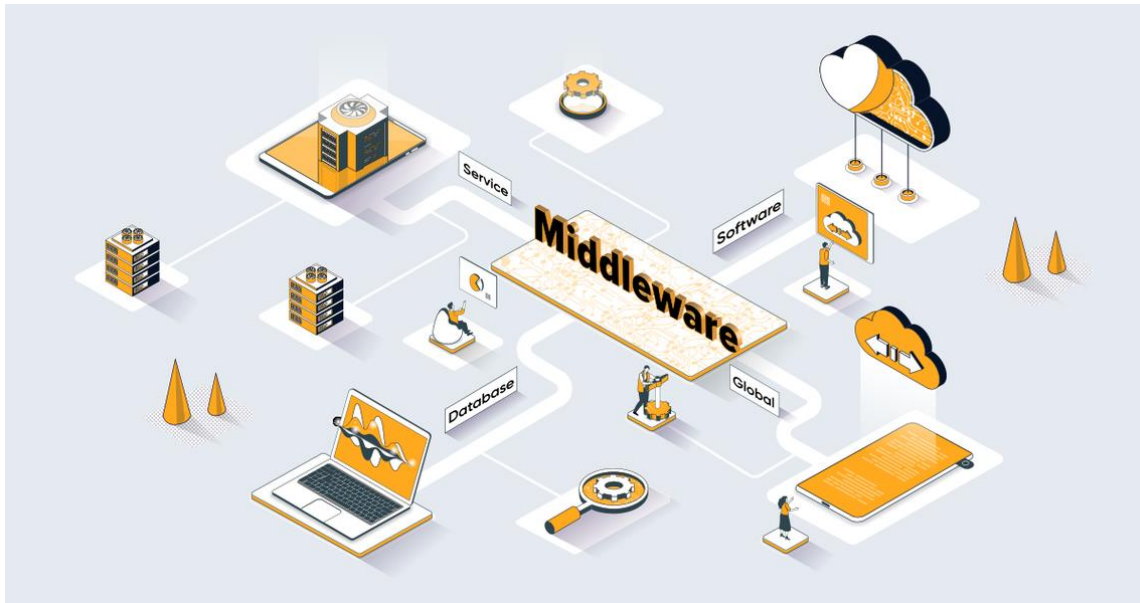


Security Design -System Design



- Identify potential security risks and vulnerabilities in the job portal system.
- Implement security measures such as authentication, authorization, and data encryption.
- Design access control mechanisms to ensure appropriate user privileges and protect sensitive data.

Integration Design -System Design



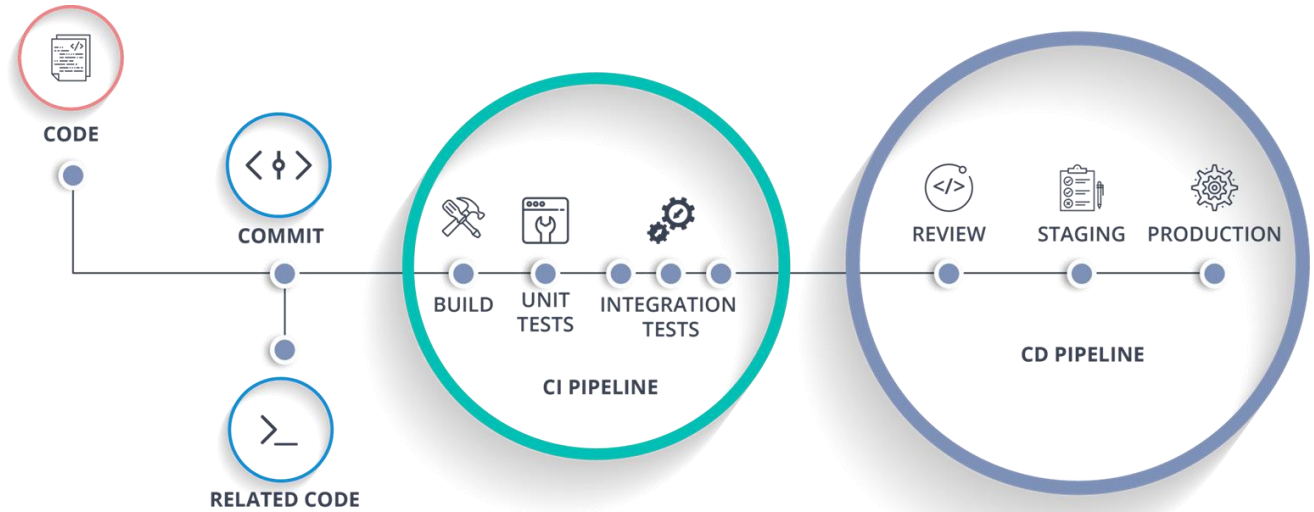
- Identify external systems or services that need to be integrated with the job portal system.
- Define the integration interfaces, protocols, and data exchange formats.
- Plan and design the integration process to ensure seamless communication between systems.

Infrastructure Design -System Design

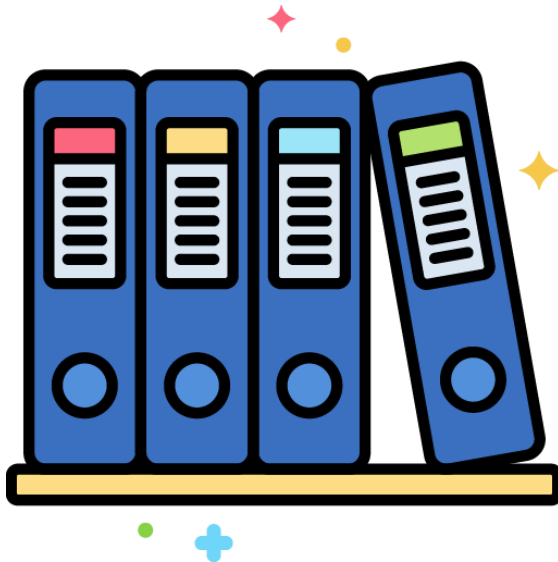
- Design the infrastructure required to host and support the job portal system.
- Determine the necessary hardware, networking components, and hosting environment (such as Azure Cloud).
- Consider factors like scalability, reliability, and disaster recovery.



Deployment Design & Plan -System Design



- Create a deployment plan outlining the steps and considerations for deploying the job portal system.
- Define the different environments such as development, testing, staging, and production.
- Incorporate a CI/CD (Continuous Integration/Continuous Deployment) pipeline for automated build, testing, and deployment.
- Outline the configuration management and release strategies for each environment.



Documentation -System Design

- Document the system design, including the architecture, UI designs, database schema, API specifications, and component specifications.
- Create a System Architecture Document that provides a comprehensive overview of the system design and its components.
- Develop detailed design documents for UI/UX, API, database, security, integration, and infrastructure.
- Include guidelines for system configuration, deployment, and maintenance.

Conclusion

- Following the System Analysis and Design process ensures a well-structured and user-centric approach to developing a complex job portal.
- The careful analysis of user needs, identification of non-functional requirements, modeling of business processes and domain (including conceptual, logical, and physical), the design of architecture, UI with a focus on UX, API, database, security, integration, infrastructure, deployment plan, and documentation collectively contribute to a successful implementation and deployment phase.

