**Content Weightages for PSHD - NHSP**

|  |  |  |
| --- | --- | --- |
| **Sr. No** | **Core Areas Division** | **% Weight** |
| 1- | Quantitative | 05% |
| 2- | General knowledge | 05% |
| 3- | Subject Trade Specific | 90% |
|  |  **Total** | **100%** |

|  |  |  |  |
| --- | --- | --- | --- |
| **Sr. No** | **Designation of the Post** | **Criteria and Subject Division** | **Subject****% Weight** |

|  |  |  |  |
| --- | --- | --- | --- |
| **01** | **Team Lead Software Development** | **Overall Percentage**  | **90%** |
| Software Development* Software Development Life Cycle
* Monolithic Architecture
* Microservices Architecture
* Differences Between Monolithic and Microservices Architecture
* Scrum Master
* Requirement Gathering and Tasking
* Git For Source Control
 | 10% |
| Database Systems* RDBMS
* Basic Queries
* Functions
* Store procedures
* Keys
* Relation’s
* Scaling of Database
* Failover and Disaster Management
* Temp Tables vs CTE
* Normalization
* Datatypes
* Schema Management
* Table Standards and Logging
 | 20% |
| Development Expertise C# , .Net Core , Angular* Difference Between .Net Framework and .Net Core
* Service Registration Life cycle in .Net Core
* Classes and Interfaces
* Couple and Decouple classes
* Constructor and Constructor Overloading
* Param, Ref, Out Modifiers
* Access Modifiers
* Primitive Types and Non-Primitive Types
* Dependency Injection
* Authentication and Authorization
* Cache / Redistributable Cache
* Entity Framework vs ADO.Net
* Load Balancing
* Middleware
 | 15% |
| Angular * Angular Modules
* Components
* Decorators
* Directives
* Life Cycle Hooks
* Communication between components
* Routing
* Data binding
* Forms types
* Lazy Loading / Pre-Loading
* Reuseable components
* Micro Frontend
* Service Worker
* Interceptor
* Services
 | 15% |
| Programming Fundamentals* Variables
* Data Types
* Basic Syntax
* Structure (Stack, Queue, Array, List)
* Iterations (Loops)
* Conditions
* Functions
 | 20% |
| Object Oriented Concepts* Classes
* Objects
* Methods
* Attributes
* Encapsulation
* Abstraction
* Inheritance
* Polymorphism
* Composition
* Coupling
 | 10% |
| **02** | **Senior Software Developer** | **Overall Percentage**  | **90%** |
| Software Development* Software Development Life Cycle
* Monolithic Architecture
* Microservices Architecture
 | 10% |
| Database Systems* RDBMS
* Basis Queries
* Functions
* Store procedures
* Keys
* Relation’s
* Temp Tables vs CTE
* Normalization
 | 20% |
| Development Expertise C# , .Net Core , Angular* Difference Between .Net Framework and .Net Core
* Service Registration Life cycle in .Net Core
* Classes and Interfaces
* Couple and Decouple classes
* Constructor and Constructor Overloading
* Param, Ref, Out Modifiers
* Access Modifiers
* Primitive Types and Non-Primitive Types
* Dependency Injection
* Authentication and Authorization
* Cache / Redistributable Cache
* Entity Framework vs ADO.Net
* Load Balancing
* Middleware
 | 15% |
| Angular * Angular Modules
* Components
* Decorators
* Directives
* Life Cycle Hooks
* Communication between components
* Routing
* Data binding
* Forms types
* Lazy Loading / Pre-Loading
* Reuseable components
* Micro Fronted
* Service Worker
* Interceptor
* Services
 | 15% |
| Programming Fundamentals* Variables
* Data Types
* Basic Syntax
* Structure (Stack, Queue, Array, List)
* Iterations (Loops)
* Conditions
* Functions
 | 20% |
| Object Oriented Concepts* Classes
* Objects
* Methods
* Attributes
* Encapsulation
* Abstraction
* Inheritance
* Polymorphism
* Composition
* Coupling
 | 10% |
| **03** | **Software Developer** | **Overall Percentage**  | **90%** |
| Programming Fundamentals* Variables
* Data Types
* Basic Syntax
* Structure (Stack, Queue, Array, List)
* Iterations (Loops)
* Conditions
* Functions
 | 30% |
| Database Systems* RDBMS
* Basic Queries
* Functions
* Store procedures
* Keys
* Relation’s
* Scaling of Database
* Failover and Disaster Management
* Temp Tables vs CTE
 | 15% |
| Object Oriented Programming* Classes
* Objects
* Methods
* Attributes
* Encapsulation
* Abstraction
* Inheritance
* Polymorphism
* Composition
* Coupling
 | 15% |
| .NET C#* Constructor and Constructor Overloading
* Param, Ref, Out Modifiers
* Access Modifiers
* Controller
* Model
* View
* Services
* DTOs
* DB Context
* String Interpolation
* Collections
* Asynchronous Task Composition
* Primitive Types and Non-Primitive Types
* Upcasting and Down casting
 | 15% |
| Angular* Role of NodeJS in Angular
* Angular Modules
* Components
* Decorators
* Directives
* Life Cycle Hooks
* Selector
* Template
* Child to Parent and Parent to Child Communication
* Routing
* Data binding
* Forms
* Lazy Loading
* Pre-Loading
 | 15% |
| **04** | **Graphic Designer** | **Overall Percentage**  | **90%** |
| Adobe Photoshop* raster images Editor
* Toolbar
* Image mode (bitmape, greyscale etc.)
* Transform image
* Color
* Layers
* Type (texts, Paragraphs)
* Select
* Filter
* View
* Alignment
 | 25% |
| Illustrator* Vector drawing Editor
* Toolbar
* Align
* Path
* Envelop
* Type (texts, Paragraphs)
* Select
* Layers
 | 20% |
| Corel Draw* Vector drawing Editor
* Toolbar
* Align
* Path
* Envelop
* Type (texts, Paragraphs)
* Select
* Power Clip
 | 20% |
| Modeling* 2D/3D Concept
* 3D software (#DS Max, Cinema4D etc.)
* Polygon
* Subdivision Surface
* Spline Shapes
* Texture Mapping
* Lights
 | 15.% |
| Rendering, and Animation* Image Resolution
* Frame Render
* External Renderers
* Keyframe Animation
* Motion Capture
* Rigging
 | 10% |
| **05** | **Senior Data Analyst** | **Overall Percentages** | **90%** |
| Database Management  | 10% |
| SPSS | 30% |
| Statistical Analysis | 10% |
| Research Methods | 10% |
| Data Structures and Algorithms | 10% |
| System Programming | 10% |
| Numerical Computing | 10% |
| **06** | **Data Analyst** | **Overall Percentages** | **90%** |
| Database Management (SQL)Basic commands such as JoinsGroup byOrder bySelect | 10% |
| Statistical AnalysisExcel (vlookup) | 10% |
| Research MethodsData sampling, population size, types of research methods | 10% |
| Report Writing Skills | 10% |
| MS-Office (Excel, PowerPoint, Word) | 30% |
| English | 20% |
|
| **07** | **Team Lead Software Quality Assurance**  | **Overall Percentages** | **90%** |
| Quality Assurance & Software Testing  | 10% |
| Software Testing Life Cycle (STLC)* Requirement Analysis
* Test Planning
* Test Case Development
* Test Environment Setup
* Test Execution
* Test Cycle Closure
 | 20% |
| Defect/Bug tracking tool:* Jira
 | 10% |
| Software Testing Types* Manual Testing

Types of Manual Testing:* White Box Testing
* Black Box Testing
* Gray Box Testing

Types of Black Box Testing* Functional Testing
* Non-Functional Testing

Types of Functional Testing* Smoke Testing
* Sanity Testing
* Regression Testing
* Adhoc Testing
* Alpha Testing
* Beta Testing
 | 30% |
| Different Levels of Software Testing:* Unit Testing
* Integration Testing
* System Testing
* Acceptance Testing
 | 20 % |
| Testing Throughout Software Life Cycle | 10% |
| **08** | **Software Quality Assurance Officer** | **Overall Percentages** | **90%** |
| Different Levels of Software Testing* Unit Testing
* Integration Testing
* System Testing
* Acceptance Testing
 | 20% |
| Software Testing Types:* Manual Testing

Types of Manual Testing:* White Box Testing
* Black Box Testing
* Gray Box Testing

Types of Black Box Testing* Functional Testing
* Non-Functional Testing

Types of Functional Testing* Smoke Testing
* Sanity Testing
* Regression Testing
* Adhoc Testing
* Alpha Testing
* Beta Testing
 | 30% |
| Software Testing Life Cycle (STLC)* Requirement Analysis
* Test Planning
* Test Case Development
* Test Environment Setup
* Test Execution
* Test Cycle Closure
 | 20% |
| Quality Assurance & Software Testing | 20% |
| **09** | **Senior Android Developer** | **Overall Percentages** | **90%** |
| Programming Fundamentals* Variables
* Functions
* Data Types
* Basic Syntax
* Structure (Stack, Queue, Array, List)
* Iterations (Loops)
* Conditions
 | 15% |
| Object Oriented Programming* Classes
* Objects
* Methods
* Attributes
* Encapsulation
* Abstraction
* Inheritance
* Polymorphism
* Composition
 | 15% |
| Android Core:* Introduction to Android Development
* Android Project Structure
* Activity and Fragments
* Intent
* Broadcast Receiver
* Content Providers
* Intents and Intent Filters
* User Interface (UI)
* RecyclerView and Adapters
* AsyncTask
* Modern Android Development (MAD) Skills
* Working with Data (SharedPreferences, SQLite, Room)
* Networking (HTTP, REST APIs, Retrofit)
* Background Tasks (Coroutines, Services, WorkManager)
* Permissions and Security

 | 30% |
| Kotlin:* Kotlin basics and syntax
* Variables and Data Types
* Operators and Expressions
* Control Flow Statements (if, when, loops)
* Arrays and Collections
* Coroutines for asynchronous programming
* Extension functions
* Null safety
* Data classes and Sealed classes
* Dependency Injection
 | 15% |
| Java: * Java program structure (classes, methods, main method)
* Variable declaration and initialization
* Type casting and type promotion
* Control Flow Statements
* Arrays and Strings
* Exception Handling
 | 15% |
| **10** | **Android Developer** | **Overall Percentages** | **90%** |
| Programming Fundamentals* Variables
* Functions
* Data Types
* Basic Syntax
* Structure (Stack, Queue, Array, List)
* Iterations (Loops)
* Conditions
 | 15% |
| Object Oriented Programming* Classes
* Objects
* Methods
* Attributes
* Encapsulation
* Abstraction
* Inheritance
* Polymorphism
* Composition
 | 15% |
| Android Core:* Introduction to Android Development
* Android Project Structure
* Activity and Fragments
* Intent
* Broadcast Receiver
* Content Providers
* Intents and Intent Filters
* User Interface (UI)
* RecyclerView and Adapters
* AsyncTask
* Modern Android Development (MAD) Skills
* Working with Data (SharedPreferences, SQLite, Room)
* Networking (HTTP, REST APIs, Retrofit)
* Background Tasks (Coroutines, Services, WorkManager)
* Permissions and Security
 | 30% |
| Kotlin:* Kotlin basics and syntax
* Variables and Data Types
* Operators and Expressions
* Control Flow Statements (if, when, loops)
* Arrays and Collections
* Coroutines for asynchronous programming
* Extension functions
* Null safety
* Data classes and Sealed classes
* Dependency Injection
 | 15% |
| Java: * Java program structure (classes, methods, main method)
* Variable declaration and initialization
* Type casting and type promotion
* Control Flow Statements
* Arrays and Strings
* Exception Handling
 | 15% |
| **11** | **Database Administrator** | **Overall Percentages** | **90%** |
| Data and Database Administration | 15% |
| Client-Server Database | 15% |
| Data Warehousing | 15% |
| Managing Multiuser Databases | 15% |
| Advanced SQL | 15% |
| Managing Databases with Oracle | 15% |
| **12** | **CCTV / Hardware Technician** | **Overall Percentages** | **90%** |
| ICT Knowledge | 30% |
| CCTV / Network Knowledge/System Hardware | 25% |
| Network Monitoring & Reporting | 25% |
| MS Windows & MS Office Knowledge | 10% |

|  |  |  |  |
| --- | --- | --- | --- |
| **Sr. No** | **Designation of the Post** | **Criteria and Subject Division** | **Subject****% Weight** |

|  |  |  |  |
| --- | --- | --- | --- |
| **13.** | **Data Centre Specialist** | **Overall Percentage**  | **60%** |
| Machine Learning & Artificial Intelligence  | 10% |
| Advance Network Architecture (STP, VPN, Tunnel, NGFW,OSPF, BGP, MPLS and SDWAN) | 10% |
| Cloud Infrastructure Architect (Azure, AWS and Hadoop) | 10% |
| Python & Dev Ops  | 10% |
| Datacenter Planning, Designing, implementation and monitoring  | 10% |
| Advance Linux and Microsoft server administration | 10% |