**Project Documentation: Automated Backup and Restoration System**

**Project Overview**

**Project Name:**

**Automated Backup and Restoration System**

**Objective:**

* Automate backup processes for system-critical files and virtual environments.
* Validate the restoration process in a virtualized environment to ensure reliability.
* Implement logging mechanisms to track all backup and restore operations.
* Ensure security and efficiency in backup storage and retrieval.
* Analyze technical problems related to customer requirements.
* Design a logical plan for the development of a technical and software solution.
* Implement technical and software solutions including documentation.
* Evaluate technical and software problems, plans, and solutions for correctness and appropriateness.
* Develop resources to meet technical requirements.
* Communicate effectively with technical and non-technical audiences.

**Scope:**

* Backup system-critical files, including system state, IIS configurations, scheduled tasks, installed programs list, event logs, drivers, Windows Defender configurations, Windows update history, and registry.
* Encrypt backups to enhance security.
* Automate retention policy to delete outdated backups.
* Validate backups through restoration checks.
* Log all backup operations for monitoring and auditing.

**Tools Used:**

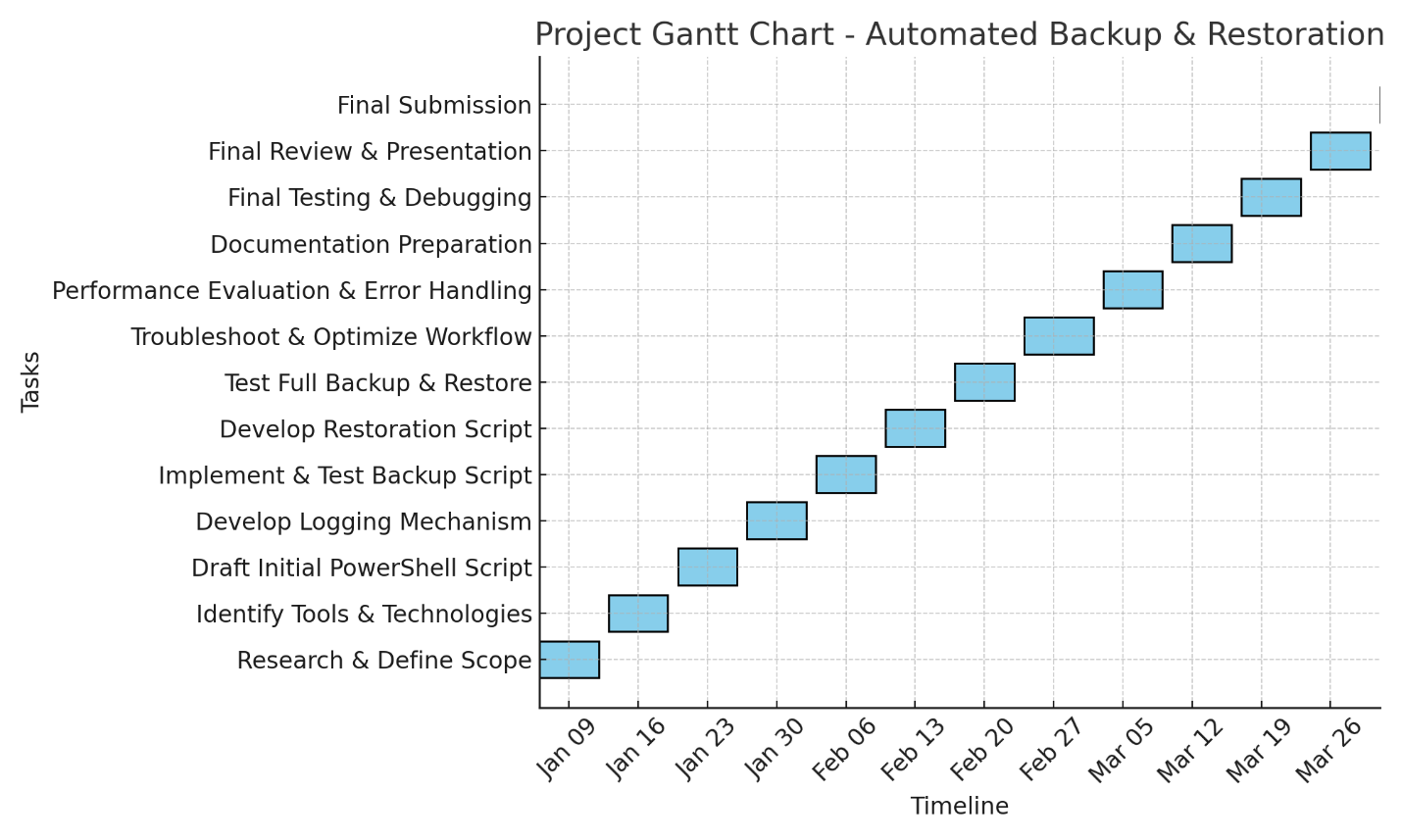
* PowerShell: For scripting and automation.
* wbadmin: For system state backup.
* xcopy: For copying files and configurations.
* Export-WindowsDriver: For driver backup.
* reg export: For registry backup.
* Protect-CmsMessage: For encrypting backups.
* Get-WmiObject: For retrieving installed programs.
* Get-HotFix: For exporting Windows update history.
* ConvertTo-SecureString: For password handling.
* Compress-Archive: For creating backup archives.

**Technical Specifications:**

* Backup Location: C:\Backups
* Encryption Key Storage: C:\BackupKey\backup\_key.txt
* Backup Retention Policy: 30 days
* Logging Directory: C:\BackupLogs

**Gantt Chart:**

|  |  |  |  |
| --- | --- | --- | --- |
| **Week** | **Date Range** | **Task** | **Status** |
| **Week 1** | Jan 6 - Jan 12 | Research and define project scope | ✅ Completed |
| **Week 2** | Jan 13 - Jan 19 | Identify tools and technologies (PowerShell, Hyper-V, Task Scheduler, etc.) | ✅ Completed |
| **Week 3** | Jan 20 - Jan 26 | Draft initial PowerShell script for file and system state backup | ✅ Completed |
| **Week 4** | Jan 27 - Feb 2 | Develop logging mechanism for tracking backup status | ✅ Completed |
| **Week 5** | Feb 3 - Feb 9 | Implement and test backup script for Hyper-V virtual machines | ✅ Completed |
| **Week 6** | Feb 10 - Feb 16 | Develop restoration script and integrate with logging system | ✅ Completed |
| **Week 7** | Feb 17 - Feb 23 | Test full backup & restore cycle in virtual environment | ✅ Completed |
| **Week 8** | Feb 24 - Mar 2 | Troubleshoot issues and optimize automation workflow | ✅ Completed |
| **Week 9** | Mar 3 - Mar 9 | Final testing, debugging, and performance evaluation | ✅ Completed |
| **Week 10** | Mar 10 - Mar 16 | Documentation and final report preparation | ✅ Completed |
| **Week 11** | Mar 17 - Mar 26 | Project submission and presentation | ✅ Completed |



**Milestones**

1. **Project Planning**

* Identify critical files and configurations for backup.
* Define security measures and encryption needs.
* Analyze technical problems as related to customer requirements.

1. **Script Development**

* Write PowerShell script for backup operations.
* Implement encryption and secure key storage.
* Automate logging mechanisms.
* Design a logical plan for the development of a technical and software solution.

1. **Testing and Validation**

* Test backup functionality.
* Validate restoration process in a virtualized environment.
* Perform error handling and debugging.
* Evaluate technical and software problems, plans, and solutions for correctness and appropriateness.

1. **Deployment and Documentation**

* Finalize script for deployment.
* Document process for future teams.
* Develop resources to meet technical requirements.
* Communicate effectively with technical and non-technical audiences.

**Initial Ideas and Final Selection Process:**

Initially, several ideas were considered for this project, including:

1. **Automated Active Directory Deployment:** Setting up an automated process to deploy Active Directory environments with predefined configurations.
2. **Home Network Security Suite:** A monitoring and firewall configuration tool for home users using open-source tools like Pi-hole and Grafana.
3. **Automated Backup and Restoration System:** A PowerShell-based solution to automate system-critical file backups and validate restorations.

After analyzing feasibility, real-world application, and technical challenges, the **Automated Backup and Restoration System** was chosen because:

* It directly addresses business and IT needs reliable data protection.
* It aligns with industry best practices for disaster recovery.
* It provides an opportunity to develop PowerShell scripting, automation, and security implementation skills.
* It has measurable outcomes and success criteria, making it ideal for evaluation.

**Screenshots**

Captura de pantalla de computadora

El contenido generado por IA puede ser incorrecto.

Imagen de la pantalla de un computador

El contenido generado por IA puede ser incorrecto.

Texto

El contenido generado por IA puede ser incorrecto.

Texto

El contenido generado por IA puede ser incorrecto.

Interfaz de usuario gráfica

El contenido generado por IA puede ser incorrecto.

**Work Breakdown Structure**

1. **Backup Automation**

* Identify system-critical files.
* Write scripts for system state, IIS, registry, event logs, and drivers backup.

1. **Security Implementation**

* Generate and store encryption keys.
* Encrypt backup files.

1. **Logging and Monitoring**

* Automate backup logs.
* Implement verification of successful backups.

1. **Backup Retention**

* Automate deletion of outdated backups.
* Maintain storage efficiency.

1. **Validation & Restoration**

* Test and verify backup integrity.
* Simulate restoration in a controlled environment.

**Challenges Encountered & Solutions**

**Issue: Missing Encryption Key Directory**

* **Problem:** Script failed when trying to write to C:\BackupKey\backup\_key.txt.
* **Solution:** Added directory creation logic to the script.

**Issue: PowerShell Cmdlet Errors**

* **Problem:** ConvertTo-Base64String and ConvertFrom-Base64String were not recognized.
* **Solution:** Replaced with [Convert]::ToBase64String() and [Convert]::FromBase64String().

**Issue: Backup Validation Failures**

* **Problem:** Encrypted backup file was not found after execution.
* **Solution:** Implemented conditional checks and logging to verify file existence.

**Issue: Retention Policy Deleting Incorrect Files**

* **Problem:** Files were deleted without checking the last modified date properly.
* **Solution:** Added logic to compare current date with LastWriteTime before deletion.

**Issue: Backup Archive Missing Before Encryption**

* **Problem:** Protect-CmsMessage failed due to missing backup file.
* **Solution:** Added compression step before encryption to ensure file existence.

**Technical Turnover to Future Teams**

* Ensure Execution with Administrative Privileges
* Monitor Log Files (**C:\BackupLogs**) for Errors
* Verify Backup and Restoration in Test Environments
* Rotate Encryption Keys Periodically for Security
* Update Retention Policy as Needed

**Scope Compliance**

**✅ Backup automation implemented**

**✅ Restoration validation included**

**✅ Logging and monitoring added**

**✅ Security measures enforced**

**✅ Retention policy configured**

**Version Control Summary**

v1.0: Initial script with basic backup functionality.

v1.1: Added encryption mechanism and logging.

v1.2: Implemented backup retention and validation.

v1.3: Debugged PowerShell errors and enhanced security.

v1.4: Finalized documentation and technical turnover procedures.

v1.5: Added backup compression before encryption to fix missing archive issue.

**Forms and Templates**

**Backup Log Template**

[Date/Time]: [Operation Performed]

Status: [Success/Failure]

Details: [Error Message (if any)]

**Issue Tracking Template**

Issue:

Date Identified:

Resolution Steps:

Status: [Open/Resolved]

**Restoration Checklist**

**✅ Validate backup file integrity**

**✅ Test restoration in a virtual machine**

**✅ Verify system configurations after restoration**

**✅ Check encryption key retrieval**

**✅ Monitor logs for errors**

**Conclusion**

This project successfully automates critical file backups, enforces encryption, validates restoration processes, and implements logging and retention policies. Future teams can further enhance it by integrating cloud storage solutions and automating restoration testing.