

**ATILIM UNIVERSITY**

Software Engineering Department

SE494

Senior Project Proposal for Cooperative Education

**<PROJECT NAME>**

Proposed by

Nazlı Nisa Öz

Moderated by

Mehmet Uğur Öney

Gardiyan Sistem Güvenlik Teknolojileri A.Ş.

Coordinated by

Ali Yazıcı

2025 - 2026 Fall

# Foreword

IMPORTANT, READ FIRST

***Do not remove or modify the sections with these notations in any way.***

*All guidelines in this template appear in this ‘boxed’ format. These instructions, as well as the preceding title page and this foreword, should never be removed from the submitted files.*

*All texts between “<” and “>” symbols (incl. on the front page and in the headers) should be replaced or removed.*

*Please comply with the formatting rules by adhering to the following guidelines:*

* *Do not remove the instructions.*
* *The proposal size should be less than 3 MB.*

# Change log

*The change log is a table which provides an overview of changes made in the Project Proposal Document. A description of the changes must be provided in a concise manner.*

|  |  |  |  |
| --- | --- | --- | --- |
| **Version** | **Submission date** | **Description of changes** | **Affected parts** |
| Draft | DD/MM/YYYY |  |  |
| v1 | 23/09/2025 |  |  |
| v2 |  |  |  |
|  |  |  |  |
|  |  |  |  |

# Table of Contents

[Foreword 2](#_Toc110845926)

[Change log 2](#_Toc110845927)

[Table of Contents 3](#_Toc110845928)

[Acronyms 4](#_Toc110845929)

[1 Project Key Data 0](#_Toc110845930)

[2 Project Description 1](#_Toc110845931)

[2.1 Project Short Description 1](#_Toc110845932)

[2.2 Projenin Kısa Tanımı 1](#_Toc110845933)

[3 Key Tools, Techniques and Technologies 2](#_Toc110845934)

[4 Project Plan 3](#_Toc110845935)

[4.1 Methodology 3](#_Toc110845936)

[4.2 Time Schedule (Gantt Chart) 4](#_Toc110845937)

[5 Deliverables List 5](#_Toc110845938)

[6 Project Management 6](#_Toc110845939)

[6.1 Risk Management 6](#_Toc110845940)

[6.2 Budget Estimation 6](#_Toc110845941)

[6.2.1 Effort Estimation (person per month) 6](#_Toc110845942)

[6.2.2 Other Project Requirements 6](#_Toc110845943)

[6.3 Standards 7](#_Toc110845944)

[6.4 Ethical and Legal Requirements / Issues 7](#_Toc110845945)

# Acronyms

|  |  |
| --- | --- |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

# Project Key Data

***Project Short Title*** *is a short name of the project, if it exists. For example “NoFaDe” or “NFDP”, for a project having Full Title of “Novel Face Detection Platform Development”.*

***Subject Classification*** *is the general Computer Engineering/Software Engineering/Information Systems Development area the project can be classified in, e.g. image recognition, software testing, Big Data, Artificial Intelligence, Web-based Application, Mobile Application, Game Application, IoT-based Smart Application, Cloud-based Application, Software Security, Cybersecurity, etc.*

|  |  |
| --- | --- |
| Project Short Title | Vigilant |
| Project Full Title | AI-Based Automated Security Scoring and Hardening |
| Subject Classification | Cybersecurity, Artificial Intelligence, Software Security, Automation |
| Student name(s)  (Team members of project) | Nazlı Nisa Öz  Kudret Yavuz Arslan |
| Full Name of Corporation | Gardiyan Sistem Güvenlik Teknolojileri A.Ş. |
| Mentor Name(s)  (Corporation member) | Mehmet Uğur Öney |
| Coordinator Name(s)  (University member) | Ali Yazıcı  Hürevren Kılıç |

# Project Description

*In this section, you are to provide information in both English and Turkish.*

*Within a maximum of one page, provide a brief description of the project that you are proposing. Your descriptions should include the problem you are addressing, the aim and the scope of your project.*

*En fazla bir sayfa içinde, projenizle ele aldığınız sorunun kısa bir tanımını yapınız. Açıklamalarınız projenizin amacını ve kapsamını içermelidir.*

## Project Short Description

The aim of this project is to develop a YARA-based rule engine that evaluates the security posture of endpoint systems and generates a comprehensive security score. The system will integrate with a Large Language Model (LLM) and the Gardiyan’s MCP server to automatically remediate detected weaknesses, ensuring adaptive, explainable, and automated endpoint security management.

The scope of the project covers the design and implementation of a rule engine using YARA, the development of a scoring mechanism for antivirus, firewall, patch management, and configuration hygiene, and the integration of this scoring system with an LLM. The LLM will be configured to interact with the MCP server to trigger corrective actions such as enabling firewalls, updating antivirus signatures, or applying critical patches.

## Projenin Kısa Tanımı

Bu proje, istemci bilgisayarların güvenlik duruşunu antivirüs durumu, firewall politikaları, yama seviyeleri ve konfigürasyon hijyeni gibi parametreleri analiz ederek değerlendiren **YARA tabanlı bir kural motoru** geliştirmeyi amaçlamaktadır. Motor, genel bir güvenlik skoru üretecek ve her bir faktör için ayrıntılı açıklama sunacaktır. Bu skor doğrultusunda, **Büyük Dil Modeli (LLM)** şirketin **MCP sunucusu** ile iletişime geçerek firewall’un etkinleştirilmesi, antivirüs imzalarının güncellenmesi veya kritik yamaların uygulanması gibi düzeltici aksiyonları başlatacaktır. Projenin kapsamı, YARA tabanlı kural motorunun tasarım ve geliştirilmesini, skorlama mekanizmasının LLM ile entegrasyonunu ve MCP üzerinden otomatik iyileştirme iş akışlarının yürütülmesini içermektedir.

# Key Tools, Techniques and Technologies

*In a maximum of two pages, outline the key tools, techniques, and technologies along with their intended use. For example;*

*- “At least three alternative Convolutional Neural Network models will be used for face identification.”*

*- Pretrained models will be used for Transfer Learning purposes.*

*- “Python language would be used to develop Jupyter Notebook files containing the solution that utilizes PyTorch Framework.”*

*- Eclipse IDE will be used in coding, debugging and execution.*

*- The Gitlab environment would be used for version tracking, Continuous Integration (CI) and Continuous Deployment (CD) purposes.*

*- etc.*

The main programming language of the project will be **Python**, which will be used to develop the orchestration logic, automation scripts, and integration modules required for endpoint evaluation and remediation. Python is chosen due to its wide adoption in cybersecurity, extensive library support, and compatibility with both rule-based detection and AI integration tasks.

Several **frameworks and libraries** will be utilized throughout the project. **PyYAML** will be employed to manage configuration files, while **Pandas** will assist in processing endpoint logs and metrics for analysis. **FastAPI** will serve as the API layer of the system, exposing rule evaluation outputs and LLM-driven decisions as web services. **LangChain** will be integrated to orchestrate interactions with the Large Language Model (LLM),

Langchain agentların sistemi

A2a agentlar arası iletişim için

Mcp agentların kullanacağı tool gibi

Yara engine geldikten sonraki skorlama nasıl olacak

Data nasıl kontrol edilecek

Datayı nasıl toplayacağız

Datalardan nasıl risk skoru çıkarılacak

The **rule engine** will be implemented using the **YARA framework**, which is widely recognized in cybersecurity for creating rules to identify and classify files, processes, and configurations based on textual or binary patterns. In this project, YARA rules will be designed not only to detect malicious artifacts but also to evaluate the **security posture** of client systems. For instance, YARA rules will check antivirus status, firewall configurations, patch levels, and other system properties. The results of these YARA evaluations will be translated into a **scoring system** that reflects the overall security hygiene of the endpoint. This approach leverages YARA’s flexibility and efficiency in pattern matching, while extending its usage from malware detection to general system security scoring.

For the **automation** layer, the system will interact with the company’s **MCP server**, which provides the necessary hooks for executing corrective actions. These include PowerShell script execution, firewall configuration adjustments, antivirus updates, and other remediation activities on the client machines. The integration ensures that once YARA-based rules detect and score a misconfiguration, corrective measures can be applied automatically.

**Version control** will be managed through Git,

Local git serverımız kullanılacak

CICD Jenkins?

with either **GitHub** or **GitLab** serving as the central repository. GitLab’s **Continuous Integration (CI) and Continuous Deployment (CD)** capabilities may also be employed for automated testing and deployment pipelines.

Project management will follow an Agile Kanban methodology, where tasks are visualized and tracked on a continuous flow basis rather than fixed sprints. For planning and progress monitoring, JIRA Kanban boards will be used to manage the backlog, assign responsibilities, and ensure smooth task transitions. This approach will provide transparency, flexibility, and continuous feedback throughout the project lifecycle.

JİRA kanbandan takip edilecek

For visualization and reporting, **Grafana** dashboards will be developed to present endpoint scores, remediation history, and live security metrics. These dashboards will allow stakeholders to monitor security posture dynamically and verify the effectiveness of remediation actions.

Önce posde proof of conceptte grafana ile yapılacak sonrasında ürünün kendi içine ekleyeceğiz

Documentation-🡪 latex , mermaid, draw.io

# Project Plan

*In this section you are to provide information about your project plan, including the methodology (e.g. Waterfall, Scrum, Kanban, DSDM, etc.) that you would be following, the Time Schedule (Gantt Chart) of your main Work Packages (or Tasks), and the Deliverable List with their due dates. Be careful that you are expected to obey the deadlines you indicate, and produce the outcomes that you list under the deliverable list.*

## Methodology

*Explain the system/software development methodology to be used such as waterfall, rapid application development, scrum, kanban, etc. Your methodology may also include a suitable Research Methodology, if appropriate for your project. Also, give information about the methodology to be used.*

In this project, the development will follow an Agile Kanban methodology. Tasks will be managed using GitHub Kanban boards, where each task will move across columns such as *To Do*, *In Progress*, *Review*, and *Done*. This approach allows the team to continuously monitor progress, prioritize new tasks dynamically, and maintain a steady flow of work without the rigid time-boxing of Scrum sprints. Since the project duration is limited to three months, Kanban offers the flexibility required to handle overlapping activities like research, design, and implementation in parallel.

The work will be divided into main Work Packages (WPs), each containing specific tasks. The focus will be on requirements gathering, design of the rule engine using YARA, development of the scoring system, integration with the LLM and MCP server, and finally validation, reporting, and deployment of the system.

\*\*\*. Each deliverable is treated as a **milestone\*\*\* dont do that**

, and the Gantt chart provided does not represent sequential Waterfall tasks, but rather the timeline of milestones aligned with Kanban’s continuous flow.

Wp, milestone ayrı

Wp—yola çıkmak

Milestone- boluda kahvaltı yapmak

Boluyu geçebilmek için kaç wp tamamlıyoruz

Wpın çıktısı etki ettiği milestonelar, langchain ile

## Time Schedule (Gantt Chart)

*Gantt Chart timing may be based on either months or weeks and you should either use Work Package or Task for the items, whichever is appropriate for your project plan. The content of the following gantt chart is given as an example and you should replace them with yours.*

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| WP /  Task No | WP /  Task Name | Start Date | End Date | Weeks | | | | | | | | | | | | | | | | | |
| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 |
| WP0 / T0 | Project Management | 01/09/2022 | 11/01/2023 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| WP1 / T1 | Requirement Analysis & Research |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| WP2 / T2 | YARA Rule Engine & System Design | 09 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| WP3 / T3 | Performance Evaluations and Model Comparison |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| WP4 / T4 | Deployment and Reporting |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| WP5 / T5 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| WP6 / T6 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| WP7 / T7 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |

Dataya karar verilmeli, research ve req analiz

Puanlama sistemi geliştirmek bir deliverable

Puanalama sistemini yara rule engine ile birleştirmek deliverable

Sonucu üreten bi sistem yazmak

Data toplayıp- uri çalıştırıp- puanlama yapması milestone

Kural çalışıyor – kural skor çıkartıyor - LLM e gidiyor - LLM agent çalıştırıyor- agent MCP çalıştırıyor- bilgisayarda bir şey düzeltiyor milestone.

# Deliverables List

*Think about the artifacts that you will produce during your proposed project and the methodology that you will follow up and list the common deliverables with due dates. The type of deliverable can be either Software, Module, Document, or Others.*

*List the intermediate and final deliverables of the project. The number and type of your reports should be determined considering this list.*

*Following table contains example items, and should be replaced with your own list.*

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **Type** | **Description** | **Due Date** |
| **D1** | Document | Graduation Project Proposal | 06/10/2025 |
| **D2** | Document | Requirement Specification & System Architecture Design | 15/10/2025 |
| **D3** | Software | |  | | --- | |  |   YARA-based Rule Engine Prototype | 07/11/2025 |
| **D4** | Document | Intermediate Progress Report | 09/11/2025 |
| **D5** | Software | |  | | --- | | Integrated LLM + MCP Automation Demo |  |  | | --- | |  | | 20/12/2025 |
| **D6** | Software | Final Version with Reporting Features | 04/01/2026 |
| **D7** | Document | Final Report | 04/01/2026 |

# Project Management

## Risk Management

*The risks that may adversely affect the success of the project and the measures to be taken to ensure the successful execution of the project when these risks are encountered (Plan B) should be outlined in the Risk Management Table. More rows can be added.*

|  |  |  |
| --- | --- | --- |
|  | **Risk Description** | **Response Action (B Plan)** |
| 1 | LLM produces unsafe/unexpected actions | Add strict policy guardrails + human approval |
| 2 | MCP integration fails | Use simulated endpoints or sandbox mode |
| 3 | Project timeline slips | Reduce scope, focus on core scoring engine |
| 4 | Lack of real endpoint data | Generate synthetic test data |

Milestonea erişememek risktir

Risk factor likelihood impact ve mitigation risk tablosu ve matriksi oluştur

## Budget Estimation

*Within this section you are to calculate the estimated cost of the project, constituting effort, and other costs (such as license, tools, libraries, etc.).*

### Effort Estimation (person per month)

*You should estimate manpower effort requirements for each Task/WP given in the Time Schedule, and sum them up to calculate total requirement in man-hour or man-month.*

* **Effort Estimation:** 2 person × 3 months = 6 person-month.
* **Other Requirements:**
  + Python IDE (available) – free
  + GitHub (available) – free
  + OpenAI API (to be purchased) –
  + Optional Cloud VM (to be purchased)

Work packagelerı çıkardıktan sonra effor estimation person per month çıkar okp?

2 kişi maaliyeti maaş. Unit olarak yazılır: 1 unit, 0.2 unit

Person-week, \*person-hour\* ile yap

### Other Project Requirements

*List any other computer, HW, SW, license, cloud service, etc. needed during project development. Specify whether the requirement is available or not. If it is not available, write its price and calculate the total cost of materials to be purchased. More rows can be added if needed.*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Id** | **Name of requirements** | **Its usage in the project (why you need it)** | **Available /  To be purchased** | **Its cost (if it needs purchasing)** |
| 1 | Danışmanlık costları  Sunucu kirası vs |  |  |  |
| 2 |  |  |  |  |
| … |  |  |  |  |
| Total | | | |  |

## Standards

*Specify the standards to be followed during the project.*

OWASP top 10 Security Standards

ISO/IEC 27001 – Information Security Management

IEEE Software Engineering Standards

## Ethical and Legal Requirements / Issues

*1. Describe any potential ethical issue that may arise during execution of the project. Describe your strategies to be followed to reduce ethical risk.*

*2. Indicate if there are any legal risks that may be encountered during the project implementation or during the commercialization of the project output.*

Ethical concern: Automated remediation may disrupt user activity.

Strategy: introduce approval thresholds and rollback options.

Legal concern: Collection of endpoint logs may include personal data.

Strategy: anonymize data and ensure GDPR/KVKK compliance.