PROJECT IDEAS · M.TECH / M.E. · CYBER SECURITY · 2026-27

M.Tech projects in cyber security, *with base papers*

These 15 M.Tech projects in cyber security cover intrusion and DDoS detection, phishing, malware, payment fraud, IoT security and LLMs for security. Each is built on an **open-access IEEE base paper** checked against its DOI record, and names the public dataset and **one measured extension** for your thesis.

Last updated 2026-11-12

Every paper below was checked against its DOI record on Crossref on 12 November 2026, and its abstract was read before its card was written, so the title, authors, journal and year are exactly as IEEE registered them. Eight of the papers also appear in our IEEE base papers list, where the cards cover B.Tech as well; here each one is scoped for an M.Tech thesis. For topics outside security, see the M.Tech project ideas.

What makes a good M.Tech project in cyber security?

Four things, settled before Stage I ends:

1. **A base paper you can reproduce.** Recent, open access, with a public dataset and enough detail in its method section to rebuild. Our guide on how to select a base paper lists what to check.
2. **A public dataset.** CICIDS2017, UNSW-NB15, NSL-KDD, ToN_IoT, CIC IoT 2023 and the others named on the cards can be downloaded today. Read each licence before you start.

3. **One measured extension.** A second dataset, an unseen attack type, a lighter model for an edge device, explanations you test, or a defence against evasion. Fix the metric and the baseline before you run anything, so the result cannot be shaped after the fact.
4. **Ethics.** Test only on data and systems you are allowed to use: public datasets, virtual machines on your own computer and test networks you build. If your project involves people, such as a survey or a phishing awareness test, get your institute's ethics approval first.

Your guide should be able to read the whole plan in one sentence: the paper you reproduce, its dataset, the one thing you change and the number that will show whether the change worked.

Which M.Tech cyber security projects can you build on an IEEE base paper?

All 15 below. Each card gives the problem and what you build and measure. Open "The base paper and the M.Tech extension" to see the paper, linked by its DOI, the data and tools, and one change to measure with the comparison table it produces.

Topic ideas, not delivered projects

Each card is built on a published, open-access IEEE paper, checked against its DOI record on Crossref. The project around it is an idea to discuss with your guide. Our 8 delivered case studies are M.Tech and M.E. projects in AI and machine learning, none of them in security yet, each shown with its real paper pages, screens and results.

Network intrusion and DDoS detection

Remove duplicate records before you split the data, and report recall for each attack class, not only overall accuracy, because the rare classes are the easiest to miss.

TOPIC IDEA · M.TECH · NETWORK SECURITY

Two-stage intrusion detection with an LSTM autoencoder

Problem: A network intrusion detector is usually tuned and tested on a single traffic dataset, so nobody knows how it behaves on traffic captured later on a different network.

Output: The paper's two-stage model, an LSTM joined to an autoencoder, rebuilt and compared with a plain DNN and a CNN on CICIDS2017 and CSE-CIC-IDS2018, with precision, recall and F1 for each attack class.

The base paper and the M.Tech extension

Base paper: A Novel Two-Stage Deep Learning Model for Network Intrusion Detection: LSTM-AE, V. Hnamte et al., IEEE Access, 2023
Data, tools and M.Tech extension: Data and tools: CICIDS2017 and CSE-CIC-IDS2018 from the Canadian Institute for Cybersecurity, the two sets the abstract names, with Python, PyTorch or TensorFlow, and scikit-learn. M.Tech extension: train on CICIDS2017, test on CSE-CIC-IDS2018 without retraining, then fine-tune on a small labelled slice of the newer set; the thesis table lists the paper's figures, your reproduction, the cross-dataset result and the fine-tuned result for each attack class.

TOPIC IDEA · M.TECH · NETWORK SECURITY

Intrusion detection for software-defined networks with CNN-BiLSTM

Problem: In a software-defined network one controller steers all the traffic, so an attacker who reaches it can reroute the whole network, and many detectors are trained on data with repeated records and badly unbalanced classes.

Output: A hybrid CNN and BiLSTM detector with the paper's feature selection stage, run as binary and multiclass classification on UNSW-NB15, NSL-KDD and the SDN-specific InSDN set, with training time recorded beside the scores.

The base paper and the M.Tech extension

Base paper: CNN-BiLSTM: A Hybrid Deep Learning Approach for Network Intrusion Detection System in Software-Defined Networking With Hybrid Feature Selection, R. Ben Said et al., IEEE Access, 2023
Data, tools and M.Tech extension: Data and tools: UNSW-NB15, NSL-KDD and InSDN, all public, and Mininet with an open controller such as Ryu if you want a small test network of your own. M.Tech extension: train on InSDN and test on attack flows you generate inside your own Mininet network, reporting macro-F1 and recall for the rarest classes; the table compares the paper's figures, your reproduction and the Mininet test for the base model and one lighter variant.

TOPIC IDEA · M.TECH · NETWORK SECURITY

Unsupervised DDoS detection and mitigation in SDN with a GAN

Problem: A DDoS flood aimed at an SDN controller can take down every service behind it, and a detector trained only on labelled attacks can miss a new kind.

Output: An unsupervised GAN built with GRU layers that checks traffic in one-second intervals and hands alarms to a mitigation step that drops malicious flows, with LSTM, convolutional and temporal convolutional versions compared on F1.

The base paper and the M.Tech extension

Base paper: An Unsupervised Generative Adversarial Network System to Detect DDoS Attacks in SDN, D. M. Brandão Lent et al., IEEE Access, 2024
Data, tools and M.Tech extension: Data and tools: CIC-DDoS2019, which is public, and a second set from the Orion group at the State University of Londrina, so check the paper for how to obtain it; Mininet and Ryu to test the mitigation in a network you build. M.Tech extension: leave one DDoS type out of training and test on it as an unseen attack, measuring F1, detection delay and the share of normal flows wrongly dropped; the table sets the GRU, LSTM and your version side by side on all three.

TOPIC IDEA · M.TECH · NETWORK SECURITY

Does an intrusion detector work on another network? A cross-dataset study

Problem: Intrusion detectors that look near perfect on the dataset they were trained on are then deployed on networks they have never seen.

Output: Four machine learning classifiers, each trained on one of four public intrusion datasets and tested on all four, with the results laid out as a train-versus-test grid and plots that show why the models fail to transfer.

The base paper and the M.Tech extension

Base paper: Machine Learning in Network Intrusion Detection: A Cross-Dataset Generalization Study, M. Cantone et al., IEEE Access, 2024
Data, tools and M.Tech extension: Data and tools: CIC-IDS-2017, CSE-CIC-IDS2018, LycoS-IDS2017 and LycoS-Unicas-IDS2018, the authors' corrected version of CSE-CIC-IDS2018 (check the paper for where it is released); scikit-learn is enough for the classifiers. M.Tech extension: test one remedy for the gap, such as extracting the same flow features with one tool for every dataset or adding domain adaptation, and show the full train-versus-test grid before and after so the examiner sees which cells improve.

Phishing, scam messages, malware and payment fraud

Handle malware samples only inside an isolated virtual machine, and collect phishing links as text without opening the pages.

TOPIC IDEA · M.TECH · FRAUD AND MALWARE

Phishing URL detection tested on links newer than the training data

Problem: Phishing pages that pose as banks, courier firms and government portals keep appearing under new addresses, and a URL model built from an old list may not recognise them.

Output: The paper's LSD model, which combines logistic regression, an SVC and a decision tree by soft and hard voting, rebuilt with canopy feature selection and grid search, and scored on precision, recall, F1 and specificity against seven single classifiers.

The base paper and the M.Tech extension

Base paper: Phishing Detection System Through Hybrid Machine Learning Based on URL, A. Karim et al., IEEE Access, 2023
Data, tools and M.Tech extension: Data and tools: the public dataset of phishing and legitimate URL attributes, covering over 11,000 websites, that the paper uses, scikit-learn, and a public phishing feed such as PhishTank or OpenPhish for fresh links. M.Tech extension: extract features for links gathered after the dataset was built, measure the fall in recall, then add domain features such as registration age and certificate details and show how much of the loss they recover; the table lists the paper's result, your reproduction, the fresh-link result and the extended model.

TOPIC IDEA · M.TECH · FRAUD AND MALWARE

Scam SMS detection for Indian messages, including Hinglish

Problem: Scam texts sent in India often mix Hindi and English and imitate banks, telecom operators and electricity boards, while most public SMS spam datasets are in English.

Output: Shallow and deep classifiers rebuilt on the paper's large SMS dataset, then tested against evasion tricks of the kind the paper studies, with the fall in detection measured for each model.

The base paper and the M.Tech extension

Base paper: Investigating Evasive Techniques in SMS Spam Filtering: A Comparative Analysis of Machine Learning Models, M. Salman et al., IEEE Access, 2024
Data, tools and M.Tech extension: Data and tools: the authors' public set of more than 68,000 messages, scikit-learn, and a multilingual transformer such as MuRIL or XLM-R from Hugging Face. M.Tech extension: build a small labelled set of Hinglish and regional-language scam texts that volunteers share with consent, with numbers and names removed, and report F1 on it for the English-trained model and for one fine-tuned on part of it; the table compares the paper's models, your reproduction and both Hinglish results.

TOPIC IDEA · M.TECH · FRAUD AND MALWARE

Android malware detection from co-existing permissions, tested on newer apps

Problem: Harmful Android apps, such as fake loan or reward apps passed around as download links, request many of the same permissions as genuine apps, so no single permission gives them away.

Output: Sets of permissions and API calls that occur together, mined with FP-growth at levels two to five, fed to classical classifiers and compared with single features on Drebin, Malgenome and MalDroid2020.

− The base paper and the M.Tech extension

Base paper: A Novel Machine Learning Approach for Android Malware Detection Based on the Co-Existence of Features, E. Odat and Q. M. Yaseen, IEEE Access, 2023
Data, tools and M.Tech extension: Data and tools: APK samples from Drebin, Malgenome and MalDroid2020, as the paper uses, with Androguard for static extraction of permissions and API calls inside an isolated virtual machine. M.Tech extension: train on the older Drebin and Malgenome apps and test on MalDroid2020, then on obfuscated copies of the test apps; the table gives F1 for a random split, the time split and the obfuscated apps, with and without co-existing features.

TOPIC IDEA · M.TECH · FRAUD AND MALWARE

Malware family classification from byte images with a Vision Transformer

Problem: Malware authors repack and lightly alter their code so each copy carries a new signature, yet samples from one family often look alike when their bytes are drawn as a greyscale image.

Output: The paper's butterfly Vision Transformer, initialised from ImageNet weights, classifying greyscale malware images by family and as malware or benign, compared with a plain ViT and a CNN on accuracy and time per image.

− The base paper and the M.Tech extension

Base paper: Global-Local Attention-Based Butterfly Vision Transformer for Visualization-Based Malware Classification, M. M. Belal and D. M. Sundaram, IEEE Access, 2023
Data, tools and M.Tech extension: Data and tools: MalImg and the Microsoft BIG malware data, which the abstract names, with PyTorch and a pre-trained ViT from timm or Hugging Face; you work with images and byte files and never run the samples. M.Tech extension: hold whole malware families out of training and measure how often they are still flagged as malware, and report CPU time per image; the table compares the CNN, the plain ViT, the paper's model and yours on seen and unseen families.

TOPIC IDEA · M.TECH · FRAUD AND MALWARE

Card payment fraud detection with an LSTM and GRU stacking ensemble

Problem: Fraudulent card and UPI payments are a tiny share of all transactions, so a model can score high accuracy while missing most of the fraud.

Output: LSTM and GRU base learners stacked under an MLP meta-learner, with SMOTE-ENN to balance the classes, compared with common classifiers on sensitivity and specificity.

The base paper and the M.Tech extension

Base paper: A Deep Learning Ensemble With Data Resampling for Credit Card Fraud Detection, I. D. Mienye and Y. Sun, IEEE Access, 2023

Data, tools and M.Tech extension: Data and tools: the abstract does not name the dataset, so read the paper's data section first; the public ULB credit card dataset on Kaggle is a common choice, and imbalanced-learn provides SMOTE-ENN. Call the work card fraud detection unless you obtain real UPI data with permission. M.Tech extension: apply SMOTE-ENN to the training data only, order the transactions by time rather than shuffling them, and report precision-recall AUC and recall at a fixed false-alarm rate; the table compares the paper's figures, your reproduction, the time-ordered result and a gradient-boosted tree baseline.

IoT and industrial network security

TOPIC IDEA · M.TECH · IOT SECURITY

Explainable intrusion detection for industrial IoT, with the explanations tested

Problem: Deep learning detectors on industrial IoT networks raise many false alarms, and a plant engineer cannot act on an alert that comes with no reason.

Output: An ensemble of deep learning models with SHAP and LIME explanations for each alert on ToN_IoT, compared with the paper's extreme learning machine baseline on precision, recall and false-positive rate.

− The base paper and the M.Tech extension

Base paper: An Explainable Ensemble Deep Learning Approach for Intrusion Detection in Industrial Internet of Things, M. M. Shtayat et al., IEEE Access, 2023
Data, tools and M.Tech extension: Data and tools: the public ToN_IoT dataset from UNSW Canberra, with the shap and lime Python packages. M.Tech extension: score the explanations, not only the detector, by removing the top features each method names and measuring how much the prediction changes, and by checking how stable those features are across five training runs; the table gives F1, false-positive rate, faithfulness and stability for each model.

TOPIC IDEA · M.TECH · IOT SECURITY

IoT botnet detection with a stacked model, then a lighter one

Problem: Cheap cameras, routers and smart plugs left on default passwords are taken over for botnets, and overall accuracy can hide poor results on the rarer attack types.

Output: The paper's ACLR model, which stacks ANN, CNN, LSTM and RNN learners, against each learner alone on UNSW-NB15, with k-fold cross-validation, ROC curves and precision-recall curves.

− The base paper and the M.Tech extension

Base paper: Hybrid Machine Learning Model for Efficient Botnet Attack Detection in IoT Environment, M. Ali et al., IEEE Access, 2024
Data, tools and M.Tech extension: Data and tools: UNSW-NB15, which labels normal traffic and nine attack categories; use its official training and testing files rather than one shuffled pool. M.Tech extension: distil the stack into one small model and measure F1 for each attack category, model size and time per flow on a Raspberry Pi; the table puts the four single models, the stack and the distilled model side by side.

TOPIC IDEA · M.TECH · IOT SECURITY

Federated IoT attack detection that resists a bad client

Problem: Hospitals, factories and housing societies each run their own IoT devices and will not hand over raw traffic, so a shared detector has to learn without pooling the data.

Output: A federated deep neural network trained across simulated clients on CIC IoT 2023, with the paper's normalisation and class balancing, compared with the same network trained centrally on the pooled data.

− The base paper and the M.Tech extension

Base paper: A Novel Federated Edge Learning Approach for Detecting Cyberattacks in IoT Infrastructures, S. Abbas et al., IEEE Access, 2023
Data, tools and M.Tech extension: Data and tools: CIC IoT 2023 from the Canadian Institute for Cybersecurity, with the Flower framework or a hand-written federated averaging loop in PyTorch. M.Tech extension: give each client only some attack types, add one client that sends poisoned updates, and test a robust aggregation rule such as the coordinate-wise median; the table reports macro-F1 for central training, plain federated averaging and robust aggregation, with and without the bad client.

LLMs and language models for security

Run open models on your own machine where you can, so no code, emails or traffic records leave it.

TOPIC IDEA · M.TECH · LLMS FOR SECURITY

SecurityBERT on an edge device: traffic read as text

Problem: A gateway in a small factory or office building has little memory and no GPU, so an attack detector that runs there must be small and quick as well as accurate.

Output: SecurityBERT rebuilt: traffic records turned into fixed-length text with the paper's privacy-preserving encoding and a byte-level BPE tokeniser, then a compact BERT that sorts traffic into fourteen attack types, with model size and CPU time per record.

The base paper and the M.Tech extension

Base paper: Revolutionizing Cyber Threat Detection With Large Language Models: A Privacy-Preserving BERT-Based Lightweight Model for IoT/IIoT Devices, M. A. Ferrag et al., IEEE Access, 2024
Data, tools and M.Tech extension: Data and tools: Edge-IIoTset, with Hugging Face Transformers and PyTorch. M.Tech extension: quantise the model to 8-bit and prune it, then measure macro-F1, size and latency on a Raspberry Pi against a CNN-LSTM baseline; the table gives macro-F1, megabytes and milliseconds per record for each version.

TOPIC IDEA · M.TECH · LLMS FOR SECURITY

LLM prompting for code vulnerability detection, with added context

Problem: Manual security review of source code is slow, static analysers miss some flaws, and LLMs that find more flaws also flag safe code as vulnerable.

Output: Self-consistency, chain-of-thought and tree-of-thought prompts run on open LLMs over labelled Java code, scored against a static analyser on precision, recall and F1.

The base paper and the M.Tech extension

Base paper: Harnessing Large Language Models for Software Vulnerability Detection: A Comprehensive Benchmarking Study, K. Tamberg and H. Bahsi, IEEE Access, 2025
Data, tools and M.Tech extension: Data and tools: the abstract says Java but does not name the test set, so read the paper's data section; the OWASP Benchmark and NIST's Juliet suite for Java are public alternatives. Serve an open model locally with Ollama and use SpotBugs or Semgrep as the static analyser. M.Tech extension: add the matching CWE description and the code that calls each function to the prompt, and measure whether precision rises without losing recall; the table compares each prompting strategy with and without the added context, beside the static analyser.

TOPIC IDEA · M.TECH · LLMS FOR SECURITY

Phishing email detection with open LLMs, including AI-written phishing

Problem: Generative AI makes fluent phishing emails cheap to produce, and small firms and colleges rarely have a security team to read suspicious mail.

Output: Base LLMs with default settings and the paper's prompt template, run on human-written and AI-generated phishing and legitimate emails, compared on accuracy and F1, with each model's explanation kept for the user.

The base paper and the M.Tech extension

Base paper: Benchmarking and Evaluating Large Language Models in Phishing Detection for Small and Midsize Enterprises: A Comprehensive Analysis, J. Zhang et al., IEEE Access, 2025

Data, tools and M.Tech extension: Data and tools: email sets of human-written and AI-generated phishing and legitimate mail as the paper describes (check its data section for the sources), open models such as Llama 3 8B Instruct run locally with Ollama, and a TF-IDF classifier as the non-LLM baseline. M.Tech extension: fine-tune the best open model with LoRA, since the paper names fine-tuning as further work, and test it on phishing emails from a source it never saw; the table compares the base model, the fine-tuned model and the TF-IDF baseline on F1 for human-written and AI-

written phishing.

Looking at payment fraud as a data science project rather than a security one? The credit card fraud detection guide explains the dataset, the metrics that matter and what each level should build.

KEY POINTS

- **15 M.Tech cyber security projects,** each built on an open-access IEEE Access paper from 2023 to 2025, checked against its DOI record.
- **Four groups:** intrusion and DDoS detection (4), phishing, malware and fraud (5), IoT and industrial security (3), and LLMs for security (3).
- **Every card names the public dataset** and one measured extension, with the comparison table your thesis would contain.
- **Test only on data and systems you are allowed to use.** Public datasets, your own virtual machines and test networks you build are enough for every idea here.
- **We quote no accuracy figures.** Read each paper's own results, then match them yourself before you claim anything.

Shortlisted a security idea?

Send the idea or its base paper, your specialisation and your stage dates. The consultation is free.

Message in the next 7 h 19 min for a reply today.

Free consultation → WhatsApp

How does an M.Tech security project run, stage by stage?

What an M.Tech cyber security project covers in Stage I and Stage II

STAGE	WORK	WHAT YOU SHOW AT THE REVIEW

STAGE	WORK	WHAT YOU SHOW AT THE REVIEW
Stage I	A literature review of recent papers on your attack type, the base paper chosen, its main result reproduced on the same dataset and split, the problem statement and the planned extension	The paper's number beside yours, and the gap your extension will test
Stage II	The extension built, the experiments run on the same test data, an ablation, the thesis and, where your guide asks for one, a paper	The comparison table, the thesis and the paper draft

Each university names and times the stages its own way, so check your own regulations.

Stage I is mostly reproduction. Follow our guide on how to implement a base paper to match the dataset version, the split and the metric, and present the result with the base paper presentation guide. In Stage II the comparison table is the heart of the thesis. Its rows are the paper's reported result, your reproduction and your extension; for security work its columns should include recall for each attack class and the false-positive rate, because a detector that raises constant false alarms is switched off in practice. The M.Tech thesis format guide shows where each part goes, and the IEEE paper format guide covers the paper.

Which security datasets should you use, and what are their weaknesses?

Use the dataset your base paper used for the reproduction, then add a second one for the extension. Every public intrusion dataset has known limits, and examiners increasingly ask about them.

Common public datasets for M.Tech cyber security projects, and what to watch for

DATASET	WHAT IT HOLDS	WHAT TO WATCH FOR
NSL-KDD	A cleaned version of the KDD Cup 1999 intrusion data, with repeated records removed	Its traffic is decades old, so it is widely criticised as unlike today's networks; use it only beside a newer set

DATASET	WHAT IT HOLDS	WHAT TO WATCH FOR
UNSW-NB15	Lab-generated traffic from UNSW Canberra with normal records and nine attack categories, and official training and testing files	Synthetic traffic, and some attack categories are rare; report results for each category
CICIDS2017 and CSE-CIC-IDS2018	Flow records with labelled attacks from the Canadian Institute for Cybersecurity	Later studies reported labelling and flow-extraction problems and released corrected versions, such as LycoS-IDS2017
CIC-DDoS2019	Traffic with many kinds of DDoS attack, also from the Canadian Institute for Cybersecurity	Lab traffic; check that your model does not just learn the attack tool's fingerprint
InSDN	Traffic captured in a software-defined network, with normal and attack flows	One test network; confirm results in a Mininet network of your own
ToN_IoT, Edge-IIoTset and CIC IoT 2023	Traffic and device data from IoT and industrial IoT test beds	Each comes from one test bed; testing across them shows whether a model generalises

Read each dataset's own page and licence before you use it, and cite the paper that describes it.

The fix examiners respect is cross-dataset testing: train on one dataset and test on another without retraining. The 2024 IEEE Access study behind the cross-dataset card found that classifiers scoring nearly perfectly on their own dataset fell close to random chance on another for most combinations of attack and dataset. A thesis that reports both numbers, and explains the gap, is stronger than one that reports a single high score.

How can we help with an M.Tech cyber security project?

We start from the base paper: we check that its dataset is public and its main result can be reproduced, then rebuild the baseline, run and write up one measured extension, and prepare the thesis chapters in your university's format, an IEEE-format paper if your guide asks for one, the review PPTs and viva preparation. Work is delivered stage by stage to match your reviews. We use public datasets and test networks you control, and we do not attack live systems or write malware.

See how it runs, what you get and the typical timeline on M.Tech projects for CSE and AI. Our delivered case studies are AI and machine learning projects; the closest to this page is network traffic prediction with a Bi-LSTM, an M.Tech project in deep learning for networks. Need only the paper from finished results? See research paper writing.

More M.Tech domains, each built on verified IEEE base papers: M.Tech projects in machine learning, M.Tech projects in image processing, M.Tech projects in embedded systems and IoT. For the full service, see our M.Tech projects for CSE and AI.

Frequently asked questions

Which cyber security topic is best for an M.Tech project?

One with a recent open-access base paper, a public dataset you can download today, and an extension you can measure in Stage II. Intrusion detection on CICIDS2017 or UNSW-NB15 is the most common and the easiest to defend; phishing, malware images and LLM topics are newer. Our guide to selecting a base paper lists the checks.

Is network security the same as cyber security for an M.Tech project?

Network security is one part of it: intrusion detection, DDoS defence and software-defined networks. Cyber security also covers malware, phishing, payment fraud and flaws in code. The intrusion, DDoS and IoT cards on this page are network security projects; the others are wider. More papers by domain are in our IEEE base papers list.

Can I do a cyber security project without attacking a real network?

Yes, and you should. Use public datasets, virtual machines on your own computer and a small Mininet network you build. Never scan a college or company network, send phishing emails to real people or run malware outside an isolated machine

unless you have written permission, because unauthorised access to computer systems can be an offence under India's Information Technology Act, 2000.

Do I need a GPU for these projects?

Not for all of them. The phishing URL, Android malware and cross-dataset projects run on a laptop. The deep learning projects fit a free cloud notebook if you train smaller models or fewer epochs. The LLM projects need a GPU or a small quantised model; check this before you commit. Our base paper implementation guide covers the set-up.

Can an M.Tech cyber security project lead to a paper?

Often, if the extension produces a clear result on the same test data as the baseline, and a well-explained negative result can count too. Acceptance is up to the reviewers of the journal or conference, so no one can guarantee it. Write it in the IEEE paper format unless your target venue gives its own template.

Can The Ultimate Project World help with an M.Tech cyber security project?

Yes. Send the idea or your base paper, your specialisation and your stage dates in the free consultation. We will tell you exactly which parts we can take on, such as the reproduction, the extension, the results, the thesis, an IEEE-format paper and viva preparation. Our M.Tech projects service explains the process.

RELATED SERVICES

From a topic to a project you can present

M.Tech projects

A base paper, one measured extension, a paper and the thesis.

IEEE projects

Start from a published base paper and extend it.

Research papers

A paper in IEEE, Springer or journal format, and a journal shortlist.