

title: "Browser Client-Side Tools for Privacy-Compliant Sequence Analysis in Teaching"

subtitle: "Architecture, Legal Assessment and Validation"

slug: "codex-vivum"

author: "Thomas Schüller"

orcid: "0009-0003-9799-6747"

email: "forschung@tschueller.com"

affiliation: "independent research, Vienna (no institutional affiliation, privately funded)"

date: "2026-05-12"

version: "v1.0"

language: "en"

license: "CC BY 4.0 (text) · AGPL-3.0-or-later (source code)"

keywords:

- bioinformatics education
- client-side tools
- local-first
- PCR primer design
- melting temperature
- restriction enzymes
- GDPR
- open science

abstract: |

Access to molecular biology and bioinformatics tools is unevenly distributed across teaching contexts. Commercial software is licensed at cost; free web tools transmit sequence data to third-party servers. This paper documents a browser client-side tool family for three routine tasks (restriction site mapping, PCR primer design, FASTA processing) in which all computation happens in the browser. Three contributions: first, positioning within a field that has grown since around 2019 (WebAssembly suites such as biowasm and ViralWasm, JavaScript libraries such as SeqViz) and identification of the remaining didactic gap; second, a differentiated legal assessment describing client-side processing as a structural but not absolute risk reduction, including delineation from medical device regulation per MDCG 2019-11 Rev. 1 (2025); third, a completed validation of the melting temperature calculation against an independent implementation of the same methodology (120 comparisons, all within ± 2 °C, constant offset of -0.89 °C attributable to the concentration convention C_T versus $C_{T/4}$). Limitations are stated: validation against experimental melting temperatures is outstanding, as is verification of restriction site detection.

1. Introduction

1.1 Problem

Access to molecular biology and bioinformatics tools is unevenly distributed in teaching. Research-intensive universities hold licences for commercial packages; schools, teacher-training colleges and institutions with limited resources regularly do not.

This inequality is empirically documented. The NIBLSE survey of 1,260 life sciences faculty (Williams, Drew, Galindo-Gonzalez et al. 2019) found lack of faculty expertise and training to be the most frequent barrier; further barriers included crowded curricula, insufficient mathematical and computational preparation of students, and limited access to resources including hardware, software and vetted teaching materials. The discrepancy between assent and practice is notable: a large majority considered integration of bioinformatics necessary, yet only about a third actually taught such content. The 2022 follow-up survey of 509 faculty (Frontiers in Education 2023) confirmed the situation; faculty at Minority-Serving Institutions and from underrepresented groups reported barriers more frequently. Access is therefore also a question of equity.

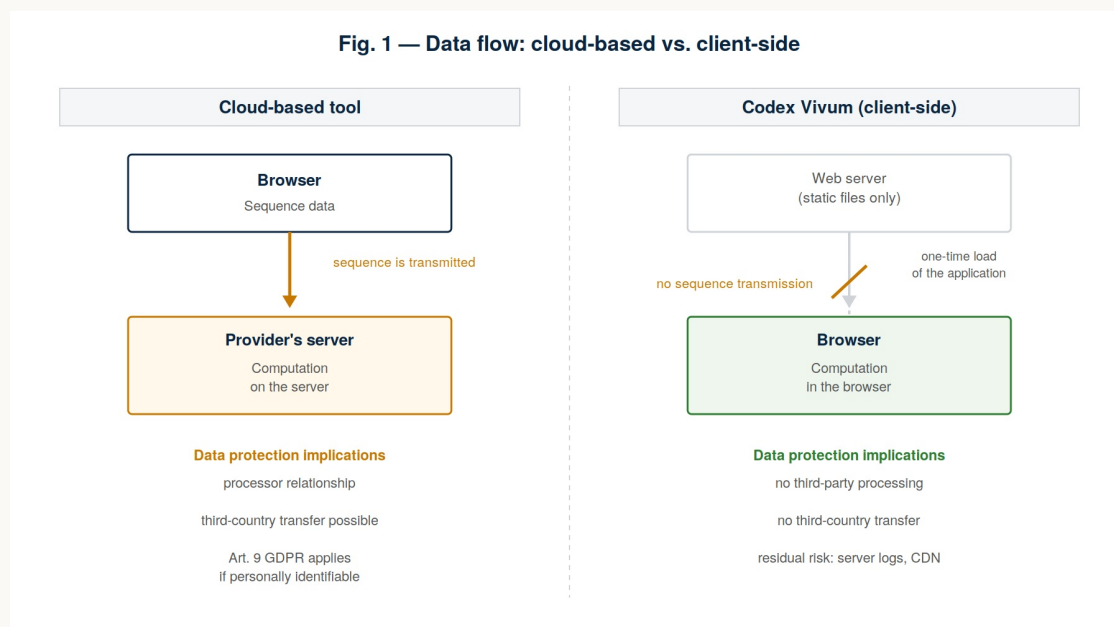
For secondary education, work such as the 4273 π project (Barker et al. 2022) shows that bioinformatics teaching in schools is feasible and curricularly connectable, provided the technical entry barrier remains low.

A data protection dimension is added. Cloud-based analysis tools transmit sequences to third-party servers. Where a personal reference exists or can be established, genetic data are special categories of personal data under Art. 9 GDPR. Independently of this, a processor relationship regularly arises, and with providers outside the European Union an additional transfer question.

1.2 Approach

Codex Vivum (Latin: *living record*) follows a deliberately narrow scope: three focused tools for routine tasks, executed entirely in the browser, without installation, without an account, without transmission of the input data.

The architecture is not merely a technical preference but a structural answer to the situation described: what is not transmitted need neither be contractually secured nor transferred to a third country (Fig. 1).



1.3 Contribution

The paper does three things: (1) positions the work within the state of the art of client-side bioinformatics and identifies the remaining gap; (2) provides a differentiated legal assessment including the limits of the approach and delineation from medical device regulation; (3) documents a completed validation of the melting temperature calculation with disclosed methodology and stated residual gaps.

2. State of the Art

2.1 Client-side tools

The field has developed markedly since WebAssembly Core 1.0 became a W3C Recommendation in December 2019. Three groups can be distinguished.

JavaScript libraries. SeqViz (Lattice Automation, MIT licence) provides a sequence viewer for DNA, RNA and protein with linear and circular display, annotations, primers, restriction sites and translation; supported formats include FASTA, GenBank, SBOL and SnapGene. Open Vector Editor (TeselaGen) offers plasmid editing, site inspection and virtual digests; its repository is now marked *deprecated*, which must be considered before adopting it as a dependency.

WebAssembly suites. biowasm compiles established C/C++ tools (samtools, bcftools, minimap2, seqtk, bedtools) to WebAssembly and makes them executable in the browser through a JavaScript interface. Building on this, ViralWasm (Ji, Aboukhalil & Moshiri 2024) describes a client-side tool suite for viral genomics. That work explicitly justifies its architecture by noting that web-based platforms would otherwise have to send potentially sensitive raw sequence data to remote servers — the same consideration underlying the approach described here. Documented limits concern compilation (32-bit, restricted SIMD instruction set), absence of JVM support, and browser memory constraints.

Server-based web tools. These must be distinguished from tools operated through the browser but computing server-side, including PlasMapper 3.0 (Wishart et al. 2023), CHOPCHOP (Montague et al. 2014; Labun et al. 2016) and SequenceServer. This distinction is decisive for data protection assessment and is frequently overlooked.

2.2 Remaining gap

Existing client-side offerings are either components for integration into one's own applications (SeqViz, Open Vector Editor) or suites with a research focus (biowasm, ViralWasm). A set of didactically reduced, immediately usable single-purpose tools deployable within one lesson — without installation, account or server component — is underrepresented. This is precisely where the tool family described here is positioned.

Open teaching infrastructures such as the Galaxy Training Network and the EMBL-EBI training portal (Swan et al. 2024) address the same access question but operate server-side and target more advanced audiences.

3. Method and Architecture

3.1 Tools

Restriction site mapping. Analysis against a library of 40 enzymes; specificities follow REBASE (Roberts et al. 2015). Functions: site detection, mapping in linear and circular DNA, fragment table for single and multiple digests, agarose gel simulation with logarithmic migration approximation, CSV export. Expansion of ambiguous recognition sequences follows IUPAC nomenclature (IUPAC-IUB 1985); the critical classes *AvaI* (CYCGRG), *ApoI* (RAATTY), *BstYI* (RGATCY) and *NlaIV* (GGNNCC) were verified.

PCR primer design. Melting temperature per the nearest-neighbor model (SantaLucia 1998) with salt correction per Owczarzy et al. (2004). Self-dimers, hairpins, pair dimers, 3' GC clamp and homopolymer runs are checked; output is a scored pair list with CSV export.

FASTA processing. Parsing, GC content, base composition, molecular weight, reverse complement, transcription, six-frame translation per NCBI Translation Table 1, ORF search above a configurable minimum length.

3.2 Architectural decisions

All computation runs in JavaScript in the browser; the web server delivers static files only. Tools are designed as standalone components and individually deployable. Libraries are bundled at build time; at runtime there are no connections to third parties. Input does not leave the browser; neither cookies nor local storage entries are created for input purposes. The generated static bundle comprises approximately 282 kB JavaScript (81 kB gzip) and 45 kB CSS (8 kB gzip).

4. Legal Assessment

4.1 Data protection

Genetic data are special categories under Art. 9 GDPR only insofar as they are personal. Sequences customary in teaching — plasmids, reference genes of model organisms, synthetic examples — regularly are not. Heightened protection would apply only once learners introduced their own, potentially identifiable sequences; for minors, Art. 8 GDPR additionally. The client-side architecture reduces this risk structurally, because such data never leave the device.

The European Data Protection Board placed guidelines on anonymisation and on processing personal data for scientific research purposes into consultation in 2026. Both are at draft stage and should be cited with that reservation.

For third-country transfers, the adequacy decision on the EU-US Data Privacy Framework (2023/1795) provides a basis; the General Court dismissed a challenge in 2025, expressly with reference to the legal and factual situation at the time of adoption. An appeal is pending. Structural uncertainty therefore remains for cloud-based tools — uncertainty a client-side architecture simply avoids.

4.2 Limits of the approach

The statement “no data transmission” applies to input data, not to operation as a whole. Three residual risks remain and should be communicated openly:

- **Server logs.** The delivering web server processes IP addresses, which case law may treat as personal data. The legal basis is regularly Art. 6(1)(f) GDPR.
- **Content delivery networks.** If libraries or fonts are loaded from third-party CDNs, data flow occurs on page load. Countermeasure: full self-hosting of all resources.
- **Third-party telemetry.** Analytics in the page context undermine the approach. Countermeasure: omission.

Subresource Integrity secures the integrity of embedded resources but is not a data protection measure.

4.3 Delineation from medical device regulation

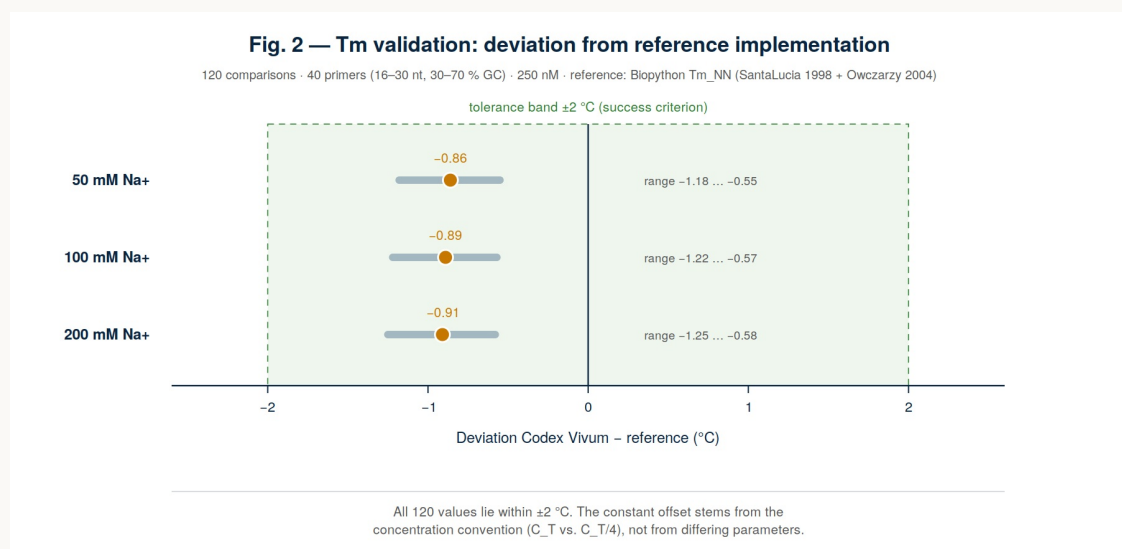
Under guidance MDCG 2019-11 as revised in 2025, software qualifies as medical device software only where it has a medical intended purpose. Tools without a diagnostic or therapeutic intended purpose fall under neither Regulation (EU) 2017/745 nor Regulation (EU) 2017/746. Since intended purpose is the decisive criterion, it is stated explicitly in the application: teaching and research only, no diagnostics, no clinical decision support. The notice appears on first use and permanently in the footer.

5. Validation of the Melting Temperature Calculation

5.1 Design

The calculation was ported from the application to Python and compared against Biopython MeltingTemp.Tm_NN — an independent implementation of the same methodology (SantaLucia 1998 with Owczarzy salt correction). Test set: 40 systematically generated primers across lengths of 16 to 30 nucleotides and GC contents of 30 to 70 per cent, each at 50, 100 and 200 mM Na⁺ and 250 nM primer concentration; 120 comparison points in total. The script is deterministic and accompanies this work.

5.2 Result



Condition	Mean deviation	Standard deviation	Max. magnitude	within ± 2 °C
50 mM Na ⁺	-0.86 °C	0.16 °C	1.18 °C	100 %
100 mM Na ⁺	-0.89 °C	0.17 °C	1.22 °C	100 %
200 mM Na ⁺	-0.91 °C	0.17 °C	1.25 °C	100 %
total (n = 120)	-0.89 °C	0.17 °C	1.25 °C	100 %

The pre-specified criterion (at most ± 2 °C in at least 80 per cent of cases) is met.

5.3 Interpreting the offset

The offset of -0.89 °C is constant across all conditions. A first run apparently yielded -1.78 °C and only 73 per cent within tolerance. Root cause analysis revealed no calculation error but a differing concentration convention: the nearest-neighbor model defines melting temperature via total strand concentration C_T , with the factor $C_T/4$ applying to non-self-complementary duplexes. Biopython represents this through two concentration parameters; its documentation notes explicitly that other tools use $C_T/4$ and that parameters must be set accordingly. After aligning the convention, the offset fell to -0.89 °C; for a single sequence parameterised to match exactly, the difference was 0.02 °C.

This finding is more than a methodological detail. Inconsistency of conventions between tools has long been documented (Panjkovich & Melo 2005; Chavali et al. 2005) and is a major source of apparent contradictions. Users comparing values from different tools must know the underlying convention. The application therefore states it.

5.4 What this validation does not show

The comparison was against an implementation of the same methodology, not against measured values. Absolute accuracy relative to experimentally determined melting temperatures is therefore untested. Public datasets exist for this step: the duplex measurement series of Owczarzy et al. (2004, 2008), the reference set of 348 sequences from Panjkovich & Melo (2005), and the Nearest Neighbor Database of the SantaLucia group.

5.5 Placing the magnitude in context

A deviation of 0.89 °C can only be judged relative to the spread that exists between the published methods themselves. On the same 20 primers, four published parameter sets give means of: Breslauer 1986 61.23 °C, Sugimoto 1996 56.41 °C, SantaLucia 1998 52.75 °C, SantaLucia & Hicks 2004 52.69 °C. The spread per sequence across these four sets averages **8.67 °C**, at most 14.63 °C.

The tool's own deviation is thus about one tenth of what the choice of parameter set alone accounts for. Two things follow. The implementation of SantaLucia 1998 is accurate within what can be determined at all. And, more important for teaching: a T_m value without a statement of method says little — anyone comparing values from two tools may be comparing Breslauer against SantaLucia and will see nearly nine degrees of difference without either being wrong. This is exactly what Panjkovich and Melo (2005) report from their method comparison. The application therefore states the method it uses.

The difference between SantaLucia 1998 and the later summary by SantaLucia and Hicks (2004) is 0.06 °C on average over the same test set; choosing the older, more widely documented set is therefore unproblematic for a teaching tool.

6. Verification of Restriction Site Detection

6.1 Design and result

The second pre-registered check concerns site detection. The comparison was against `Bio.Restriction` from Biopython [cock2009], whose enzyme data derive from REBASE. Test set: 19 sequences — twelve generated sequences across lengths of 1 to 10 kb at GC contents of 35 to 65 per cent, five constructed edge cases (site at the start, at the end, across the seam of circular DNA, closely spaced and ambiguous patterns), and the cloning vectors pUC19 (2,686 bp, GenBank L09137) and pBR322 (4,361 bp, J01749.1). All 43 enzymes in the library were checked, for linear and circular DNA: 1,634 position comparisons and 178 comparisons of fragment sizes.

Added to this is a comparison against a third source independent of the reference implementation: for pUC19 it is published that the ten polylinker enzymes `EcoRI`, `SacI`, `KpnI`, `SmaI`, `BamHI`, `XbaI`, `SalI`, `PstI`, `SphI` and `HindIII` each cut exactly once. The tool finds exactly one site for all ten.

After fixing the defect described in section 6.2, all 1,634 positions and all 178 fragment sizes agree with the reference. The three enzymes with ambiguous recognition sequences — `AvaI`, `BstEII`, `HinfI` — agree on every test sequence.

6.2 A defect found

The first run yielded 99.18 per cent agreement. All deviations concerned `NlaIV`, recognition sequence `GGNNCC`; the tool systematically found fewer sites than the reference. Inspection of the differing positions showed they were valid matches: at sequences such as `GGGTCCC` two recognition sites overlap, and the regular expression continued past the end of a match rather than one position onward. The second match was lost.

The defect manifested not as a malfunction but as the silent absence of individual cut sites — with correspondingly wrong fragment lengths. A systematic test showed that seven of the 43 patterns can overlap themselves (`ClaI`, `NheI`, `NotI`, `NsiI`, `PvuI`, `SphI`, `NlaIV`); for the six unambiguous patterns, overlap is rare enough in random sequences that it did not occur in the test set. The defect was latent there.

It was fixed by wrapping the search pattern in a lookahead that consumes nothing, with the scan advancing by one position rather than by the match length.

Continuing the check revealed two further problems. First, the validation script kept a hand-copied second version of the enzyme table; comparison with the source revealed nine discrepancies. Enzymes were checked that do not exist in the tool, while nine present ones were never covered — the figure initially reported therefore referred to a library that does not exist in that form. The script now reads the table from the source and aborts if it is missing.

Second, the run over the actual library yielded 99.04 per cent, with deviations exclusively at `BsaI`: every bottom-strand recognition site was off by exactly four nucleotides. `BsaI` is a type IIS enzyme and cuts outside its recognition sequence, per REBASE `GGTCTC(1/5)` — one nucleotide beyond the site on the top strand, five on the bottom. The data model knew only one cut offset; bottom-strand sites require the second. Fixed by an additional field in the enzyme entry and separate arithmetic for type IIS enzymes.

These three findings are the real yield of the check. None would have surfaced without comparison against an independent implementation; the second concerned not the tool but the check itself, and the third would have stayed hidden without fixing it, because `BsaI` was absent from the hand-copied list. They are documented here because a validation that reports only confirmations misses its purpose.

6.3 Limits of this check

The reference draws its enzyme data from the same source as the tool. Agreement evidences correct use of that data, not its correctness. Positions and fragment lengths were checked, not the gel display and not the overhang information. Testing used generated sequences; a comparison on annotated vectors such as pUC19 against NEBcutter remains worthwhile. Methylation dependence, as for DpnI, is not modelled by the tool.

6.4 Verification of enzyme metadata

Besides positions, the application displays two pieces of information that enter no calculation but are read off: the type of overhang and the isoschizomers. Both were checked against the same reference. The overhang type agrees for all 43 enzymes. Among the isoschizomers a contradiction emerged: PciI (ACATGT) was listed as an isoschizomer of AflIII, which however recognises ACRYGT — four sequences, of which ACATGT is only one. Enzymes whose recognition sequence forms a proper subset of another's are not isoschizomers; per REBASE the isoschizomers of PciI are BspLU11I and PscI. The entry was corrected.

The case is instructive because it would have escaped the previous logic of checking: it changes no computed position, only a displayed relationship. Anyone concluding that both enzymes cut the same sites would plan a digest wrongly on a sequence containing ACGCGT — without the application showing anything untoward. The source organism entries are unchecked; the reference does not carry this field in comparable form.

7. Limitations

- No validation against experimental melting temperatures (Section 5.4).
 - Restriction sites were verified against an independent implementation and against the published map of pUC19 (section 6), not against NEBcutter itself.
 - No sequence alignment against genome databases: such searches require server-side indices and are not representable with a purely client-side architecture. NCBI BLAST and Ensembl are referenced instead.
 - References to primary literature are maintained manually; there is no automated feed.
 - Use requires a current browser — relevant in teaching precisely where equipment is scarce.
 - Parameters follow SantaLucia (1998); later summaries (SantaLucia & Hicks 2004) were not incorporated.
- For teaching purposes this is a deliberate simplification, not an accuracy claim.

8. Positioning

8.1 Local-first as a position

Kleppmann et al. (2019) formulate with *local-first software* a set of goals including offline capability, user data ownership and longevity independent of a provider's continued existence. The architecture described here is an application of that position in education: it makes data protection a property of the construction rather than a matter of contract.

8.2 Open teaching tools as commons

Hess and Ostrom (2007) describe knowledge as a commons; the subsequent framework for governing knowledge commons (Frischmann, Madison & Strandburg 2014) transfers Ostrom’s analytical approach to data sharing and software collaboration. The defensible core for this project: knowledge itself is non-rival, the *infrastructure* of its maintenance — hosting, curation, documentation, licence upkeep — is not. Open teaching software is therefore sustained not by its licence but by governed maintenance. That is less a metaphor than a requirement on how to proceed.

8.3 On the limits of evolutionary analogies

Analogies between software ecosystems and ecological or evolutionary mechanisms (niche formation, selection, redundancy) suggest themselves and are frequently used. A review of the literature shows, however, that they remain largely illustrative there and are rarely operationalised empirically. They are therefore not carried as a load-bearing argument here. The commons-theoretical frame is more robust for the question at hand.

9. Publication and Reproducibility

Licences. Text and documentation are under CC BY 4.0, source code under AGPL-3.0-or-later. The network clause of the AGPL obliges anyone offering the software as a service to disclose the source code including modifications. As sole rights holder, the author may additionally grant divergent licences. Should third-party contributions be accepted, a prior contributor agreement is required; without one, later divergent licensing is practically foreclosed.

The obvious alternative — a licence of the form “free for private use, paid for commercial use” — was examined and rejected. Three reasons. First, point 6 of the Open Source Definition forbids restrictions by field of endeavour; such a licence would not be open source, and software-specific journals uniformly require a recognised one. Second, the term “non-commercial” is demonstrably imprecise: the study commissioned by Creative Commons [ccnc2009] found more uncertainty than clarity outside advertising and direct monetary gain, and a German appellate court read that ambiguity expressly against the licensor (OLG Köln, judgment of 31 October 2014, case 6 U 60/14). Third, the protective purpose does not apply here: licences such as the Business Source License fend off a third party marketing server software as its own service — with an application that has no server backend, that risk does not arise.

What follows is a three-tier model: private and institutional use free of charge under the AGPL; payment only for *operating* the service and for release from the disclosure obligations. The advantage is not merely legal — it never has to be decided whether a use is “commercial”. The question on which non-commercial clauses founder does not arise.

Citability. A CITATION.cff per the Citation File Format and Zenodo metadata accompany this work. For archiving, the separation of version-specific and concept DOI as provided by Zenodo is intended. The orientation follows the FAIR Principles for Research Software (Chue Hong et al. 2022; Barker et al. 2022) and the Software Citation Principles (Smith et al. 2016).

Publication route. For a submission to the Journal of Open Source Software it must be noted that web-based tools are only conditionally in scope there; a testable core library separated from the interface and automated tests are required in particular. Both are now in place (see Reproducibility). Alternatives with a different scope are BMC Bioinformatics (Software Article) and F1000Research.

Reproducibility. The validation scripts are deterministic (fixed random seeds) and are included with their

dependency specifications. The application itself processes no personal or sensitive data.

Source layout. The computational logic is kept separate from the interface in dedicated modules (`src/lib/restriction.js`, `src/lib/thermo.js`) and consists of pure functions with no reference to the display layer. A test suite of 44 checks builds on this and runs without additional software (`node --test`). Three of these checks pin down the defects described in section 6 so they cannot return unnoticed. This separation is also the precondition for a tool submission to the Journal of Open Source Software being in scope.

10. Conclusion

The contribution of this work lies less in functional scope than in framing: no server, no account, no runtime dependency on third parties, but disclosed methodology and stated limits. The validation shows that the calculation agrees with an independent implementation of the same methodology — and equally, how readily a difference of convention can appear to be an error.

The next steps are determined: validation against experimental measurements and separation of core library and interface. Verification of restriction sites has been carried out and brought a defect to light that would otherwise have gone unnoticed. Until then the statement documented here holds to exactly the extent it is evidenced, and no further.

Appendix A — Operation

The bundle contains a build setup (Vite) as well as a pre-generated static directory that can be transferred directly to a web server without a local toolchain. The base path is configurable. Self-hosting of all resources without third-party CDN is recommended.

Appendix B — Statement of Intended Purpose

This application is not a medical device within the meaning of Regulation (EU) 2017/745 and not an in-vitro diagnostic within the meaning of Regulation (EU) 2017/746. It serves teaching and research purposes only. No medical diagnosis, no therapeutic recommendation, no clinical decision support. Applicable law governs laboratory work at the user's own responsibility. Liability excluded to the extent permitted by law.

Appendix C — Disclosure

Production. The tools, validation scripts, figures and text of this work were produced with the assistance of an AI system (Claude, Anthropic). Conception, substantive decisions, verification and responsibility rest with the author.

This statement appears here because it matters for how the work is to be read: a text co-written by a system capable of producing plausible material without having verified it must meet stricter standards of evidence than one drawn from verified expertise. That is precisely why every claim in this work is tied to a traceable source or an executable validation script, and every limitation is stated.

Verifiability. The three validations are included as scripts, are reproducible via fixed random seeds, and exit with status zero only on complete agreement. The test suite comprises 46 checks with no external dependencies. Anyone doubting the claims of this work can run them.

Four defects found. The checks brought four defects to light: lost overlapping recognition sites, an enzyme table duplicated in the validation script, an incorrect cut offset for type IIS enzymes on the bottom strand, and an incorrect isoschizomer entry. All have been fixed and pinned down by tests. They are named because their existence says something about the reliability of the procedure — and because a work reporting only confirmations deserves less trust, not more.

Full disclosure, including the list of what is expressly not evidenced, in the accompanying document `OFFENLEGUNG.md`. —

References

1. Barker D., Alderson R.G., McDonagh J.L., Plaisier H., Comrie M.M., Duncan L., Muirhead G.T.P., Sweeney S.D. (2022). Bringing bioinformatics to schools with the 4273 π project. *PLOS Computational Biology* 18(1): e1009705. DOI: 10.1371/journal.pcbi.1009705
2. Barker M., Chue Hong N.P., Katz D.S. et al. (2022). Introducing the FAIR Principles for research software. *Scientific Data* 9: 622. DOI: 10.1038/s41597-022-01710-x
3. Chavali S., Mahajan A., Tabassum R., Maiti S., Bharadwaj D. (2005). Oligonucleotide properties determination and primer designing: a critical examination of predictions. *Bioinformatics* 21(20): 3918–3925. DOI: 10.1093/bioinformatics/bti633
4. Chue Hong N.P. et al. (2022). FAIR Principles for Research Software (FAIR4RS Principles) 1.0. Research Data Alliance. DOI: 10.15497/RDA00068
5. Frischmann B.M., Madison M.J., Strandburg K.J. (eds.) (2014). *Governing Knowledge Commons*. Oxford University Press.
6. Hess C., Ostrom E. (eds.) (2007). *Understanding Knowledge as a Commons: From Theory to Practice*. MIT Press.
7. IUPAC-IUB Joint Commission on Biochemical Nomenclature (1985). Nomenclature for incompletely specified bases in nucleic acid sequences. *Nucleic Acids Research* 13(9): 3021–3030. DOI: 10.1093/nar/13.9.3021
8. Ji D., Aboukhalil R., Moshiri N. (2024). ViralWasm: a client-side user-friendly web application suite for viral genomics. *Bioinformatics* 40(1): btae018. DOI: 10.1093/bioinformatics/btae018
9. Kleppmann M., Wiggins A., van Hardenberg P., McGranaghan M. (2019). Local-first software: you own your data, in spite of the cloud. *Proceedings of the 2019 ACM SIGPLAN International Symposium on New Ideas, New Paradigms, and Reflections on Programming and Software (Onward!)*, 154–178.
10. Labun K., Montague T.G., Gagnon J.A., Thyme S.B., Valen E. (2016). CHOPCHOP v2. *Nucleic Acids Research* 44(W1): W272–W276. DOI: 10.1093/nar/gkw398
11. Medical Device Coordination Group (2025). MDCG 2019-11 Rev. 1 — Guidance on Qualification and Classification of Software in Regulation (EU) 2017/745 and Regulation (EU) 2017/746.
12. Montague T.G., Cruz J.M., Gagnon J.A., Church G.M., Valen E. (2014). CHOPCHOP. *Nucleic Acids Research* 42(W1): W401–W407. DOI: 10.1093/nar/gku410
13. Owczarzy R., You Y., Moreira B.G., Manthey J.A., Huang L., Behlke M.A., Walder J.A. (2004). Effects of sodium ions on DNA duplex oligomers: improved predictions of melting temperatures. *Biochemistry*

14. Owczarzy R., Moreira B.G., You Y., Behlke M.A., Walder J.A. (2008). Predicting stability of DNA duplexes in solutions containing magnesium and monovalent cations. *Biochemistry* 47(19): 5336–5353. DOI: 10.1021/bi702363u
15. Panjkovich A., Melo F. (2005). Comparison of different melting temperature calculation methods for short DNA sequences. *Bioinformatics* 21(6): 711–722. DOI: 10.1093/bioinformatics/bti066
16. Panjkovich A., Norambuena T., Melo F. (2005). dnaMATE: a consensus melting temperature prediction server for short DNA sequences. *Nucleic Acids Research* 33: W570–W572. DOI: 10.1093/nar/gki379
17. Roberts R.J., Vincze T., Posfai J., Macelis D. (2015). REBASE — a database for DNA restriction and modification: enzymes, genes and genomes. *Nucleic Acids Research* 43: D298–D299. DOI: 10.1093/nar/gku1046
18. SantaLucia J. (1998). A unified view of polymer, dumbbell, and oligonucleotide DNA nearest-neighbor thermodynamics. *PNAS* 95(4): 1460–1465. DOI: 10.1073/pnas.95.4.1460
19. SantaLucia J., Hicks D. (2004). The thermodynamics of DNA structural motifs. *Annual Review of Biophysics and Biomolecular Structure* 33: 415–440. DOI: 10.1146/annurev.biophys.32.110601.141800
20. Smith A.M., Katz D.S., Niemeyer K.E., FORCE11 Software Citation Working Group (2016). Software citation principles. *PeerJ Computer Science* 2: e86. DOI: 10.7717/peerj-cs.86
21. Swan A., Broadbent J., Gurwitz K. et al. (2024). Making bioinformatics training FAIR: the EMBL-EBI training portal. *Frontiers in Bioinformatics* 4: 1347168. DOI: 10.3389/fbinf.2024.1347168
22. Regulation (EU) 2016/679 (General Data Protection Regulation). OJ EU L 119/1.
23. Williams J.J., Drew J.C., Galindo-Gonzalez S. et al. (2019). Barriers to integration of bioinformatics into undergraduate life sciences education. *PLOS One* 14(11): e0224288. DOI: 10.1371/journal.pone.0224288
24. Williams J.J. et al. (2023). Revisiting barriers to implementation of bioinformatics into life sciences education. *Frontiers in Education* 8: 1317191. DOI: 10.3389/educ.2023.1317191
25. Wishart D.S. et al. (2023). PlasMapper 3.0 — a web server for generating, editing, annotating and visualizing publication quality plasmid maps. *Nucleic Acids Research* 51(W1): W459–W467. DOI: 10.1093/nar/gkad276
26. Creative Commons / Netpop Research (2009). *Defining Noncommercial: A Study of How the Online Population Understands Noncommercial Use*. Creative Commons Corporation.
27. Open Source Initiative. *The Open Source Definition*, point 6 (No Discrimination Against Fields of Endeavor).
28. Oberlandesgericht Köln, judgment of 31 October 2014, case 6 U 60/14.