

Developability Assessment for De Novo Designed Binders

A sequence-level triage pass to run before you order DNA: the liabilities that actually kill generated designs, how to read published thresholds without over-trusting them, and the open-source tools to build your own screen.

THE ARGUMENT

Generative design made binder sequences cheap. Triage did not keep up. The metrics that rank a design pool are about binding, so a design can reach DNA synthesis without a developability pass having been run at all, and the cost shows up later as expression failure, aggregation and unusable panels.

CONTENTS

01	The gap nobody screens for	06	Using published thresholds honestly
02	A different bar than a clinical antibody	07	Open-source tooling and licences
03	The liabilities that kill designs	08	Run your own sequence
04	Where sequence alone runs out	09	Where Ranomics fits
05	A triage workflow before synthesis		

The gap nobody screens for

A design can score beautifully on predicted affinity and still be undevelopable. That combination used to be rare, because sequences were expensive. It is now a common failure mode.

Deep-learning design changed the economics at the front of the pipeline. RFdiffusion generates protein backbones for a specified target site (Watson et al., 2023), and BindCraft reports functional binders from a single design campaign without experimental optimisation (Pacesa et al., 2025). A lab that could previously afford to test a handful of candidates can now produce hundreds of plausible ones. The in-silico metrics that rank them are almost entirely about binding: predicted interface confidence, predicted aligned error, rough shape complementarity.

That gap has been measured. Khakerwala et al. (2025) took 538 RFdiffusion designs against SARS-CoV-2 nucleocapsid, produced 19, and found all of them soluble but 13 unexpectedly oligomeric, and report that their predictive metrics did not align with the experimental result. The designs were not bad at folding. They were not screened for the property that decided the outcome.

None of those metrics ask whether the molecule can be made. Nothing in an affinity score notices that the designed helix face is a solvent-exposed hydrophobic stripe, that the model placed an asparagine two residues upstream of a serine and so created a glycosylation sequon, or that the sequence carries an odd number of cysteines. The generator was never asked. So the flag never appears, and the design goes to synthesis.

What the failure looks like at the bench

The bill arrives weeks later, usually as ambiguity rather than as a clean negative result:

- Low or no soluble expression, which reads as a bad design rather than a bad sequence.
- Material that expresses but aggregates on concentration, so the binding assay runs on whatever stayed in solution.
- Heterogeneous product from unintended N-linked glycosylation in a eukaryotic host, confounding every downstream measurement.
- Disulfide scrambling and dimer laddering from an unpaired cysteine.
- Apparent binding that turns out to be non-specific stickiness. This costs the most, because it looks like success.
- A hit that cannot be reformatted, stored or scaled, discovered after it has become the lead.

Most of those leave a sequence-level signal that a triage pass would have raised before the order was placed, and section four is honest about the ones it would not. A triage pass is cheap: it runs on the sequence you already have, in seconds, with no structure and no wet-lab work. The reason it gets skipped is not cost. It is that generative pipelines do not include it by default, so the reader of a design table has no column that would show it.

THE POINT OF THIS GUIDE

Not to add a gate that rejects designs. Generated binders are cheap, and a screen that discards aggressively will throw away good molecules. The point is to know, before you commit synthesis budget, which of your designs carry which liabilities, so that the panel you order is a deliberate choice rather than an accident.

A different bar than a clinical antibody

Most published developability practice was built for monoclonal antibodies heading into the clinic. A de novo minibinder at the design stage is a different molecule with a different job, and applying the clinical checklist verbatim gets both the severity and the priorities wrong.

The molecule is not an IgG

A designed minibinder is frequently a small single-domain scaffold: a few dozen to a couple of hundred residues, often all-helical, with no light chain, no Fc, sometimes no disulfide and no canonical CDR loops. Framework-anchored reasoning does not transfer. A metric defined on IgG variable domains, such as total CDR length or heavy-versus-light chain charge asymmetry, has no value for a designed helical bundle: the features it is defined over do not exist there.

The stage is earlier, so the question is narrower

Clinical developability asks whether one molecule survives manufacture, formulation at high concentration and years on a shelf. Design-stage developability asks which of two hundred candidates deserve synthesis capacity, and which carry a liability you should know about before you read the assay. Viscosity is not your problem yet. Soluble expression, specificity and reformattability are.

Fixing is cheap, because the sequence is not yet a commitment

This asymmetry matters most and is missed most often. At clinical stage, removing a liability means re-engineering a validated molecule and defending the change. At design stage the sequence is a draft: most liabilities come out with a conservative point substitution and a re-fold, or by re-running the generator with a constraint, at the cost of compute rather than months. Fixing a glycosylation sequon before you order costs one substitution. Finding it after expression costs the batch and the interpretation of every assay run on it.

What carries over, and what does not

CONCERN	AT DESIGN STAGE	WHY
Unpaired cysteine	High	Breaks expression and purification in most hosts, and is trivially fixable.
N-linked glycosylation sequon	High	Creates heterogeneity that confounds every downstream measurement in a eukaryotic host.
Exposed hydrophobic and aggregation-prone stretches	High	A common cause of poor soluble yield, and a direct cause of non-specific binding.
Polyreactivity and non-specific stickiness	High	Produces false positives, so it corrupts the selection and not just the molecule.
Extreme pI or net charge	Medium	Affects solubility and purification, but the acceptable window is wide and context-dependent.
Deamidation and isomerisation hotspots	Medium, position-dependent	Matters most in or beside the binding surface, where chemical change alters function.
Humanness and immunogenicity risk	Low now, decisive later	Irrelevant to a research reagent, gating for a therapeutic. Cheap to note, expensive to retrofit.
High-concentration viscosity, freeze-thaw, shelf life	Out of scope	Needs purified material and formulation work. Not answerable from a sequence.

That ordering is a design-stage ordering and it reflects one criterion: how much a liability distorts the experiment you are about to run. Anything that makes your assay unreadable outranks anything that only makes the molecule harder to develop later.

The liabilities that kill designs

Each entry: what it is, why generative design tends to produce it, how it shows in the sequence, and what to do.

N-linked glycosylation sequons N-X-S/T, X not P

WHAT IT IS An acceptor site for N-linked glycan attachment. In a eukaryotic host it can be occupied, partly occupied or unoccupied, turning one designed sequence into a mixture of glycoforms.

WHY DESIGNS GET IT Asparagine, serine and threonine are all favoured polar surface residues. A designer optimising for a soluble, well-packed surface creates the pattern by chance, with no penalty term telling it not to.

SEQUENCE SIGNAL A literal three-residue scan. Rank hits by solvent exposure and by proximity to the binding surface, since a sequon on the paratope is a functional problem and not only a heterogeneity problem.

WHAT TO DO Substitute one of the three positions. Asparagine to glutamine or aspartate is usually least disruptive; serine or threonine to alanine is the alternative when the asparagine is load-bearing. Re-fold and confirm the interface survived. Bacterial expression does not carry the eukaryotic N-linked glycosylation machinery, so a sequon stays silent in *E. coli* and only becomes a question when the molecule is reformatted into a eukaryotic host. Record it either way.

Unpaired cysteines odd cysteine count

WHAT IT IS A free thiol with no partner. It drives intermolecular disulfides, covalent dimers and higher multimers, scrambling during folding, and oxidation-linked heterogeneity.

WHY DESIGNS GET IT Two causes. A generator working from a cysteine-containing scaffold can drop one member of a native pair while keeping the other. And a sequence designer with cysteine in its alphabet will place one for packing reasons, with no notion that thiols need partners.

SEQUENCE SIGNAL Count cysteines. An odd count means at least one unpaired thiol. An even count proves nothing: two cysteines too far apart or wrongly oriented to bond are two free thiols, so check the distance in the predicted model as well as the count.

WHAT TO DO Usually a clean fix. Mutate the odd cysteine to serine or alanine and re-fold, or restore the missing partner. For a binder with no functional need for a disulfide, excluding cysteine from the design alphabet avoids the problem at source.

Hydrophobic and aggregation-prone stretches

WHAT IT IS A contiguous run of residues hydrophobic or aggregation-prone enough to nucleate self-association when exposed to solvent. A common cause of poor soluble yield, precipitation on concentration, and non-specific binding.

WHY DESIGNS GET IT Structural rather than accidental. Designed binders bury hydrophobic surface at the interface because that is what drives affinity, and the residues that make a good hydrophobic paratope are the ones that make a bad exposed patch. The unbound state, where that surface is exposed, is often not in the objective.

SEQUENCE SIGNAL Two complementary scans. A windowed average on the Kyte-Doolittle hydrophathy scale (Kyte and Doolittle, 1982) locates sustained hydrophobic runs. An aggregation-propensity scale such as the one underlying AGGRESCAN, derived from in vivo experiments (Conchillo-Solé et al., 2007), locates stretches whose composition favours aggregation rather than merely being greasy. A buried core lights up both and is not a liability.

WHAT TO DO Separate exposed from buried before acting: this is the class where sequence scanning produces the most false positives. For an exposed patch away from the interface, substitute to polar or charged residues and re-fold. For one that is part of the paratope you are trading affinity against behaviour, which is a decision and not a fix: carry it with a watch and build an aggregation control into the first expression.

Charge and pI extremes

WHAT IT IS An isoelectric point close to the working pH leaves the molecule near net neutral, with the least electrostatic repulsion resisting self-association. Strong net charge in either direction drives non-specific electrostatic binding and complicates ion-exchange purification.

WHY DESIGNS GET IT Net charge is rarely a design objective, so the pI is whatever the composition gives. Designers that favour charged residues to improve predicted solubility can push a small binder to an extreme: in a hundred-residue molecule a handful of residues moves it a long way.

SEQUENCE SIGNAL Theoretical pI and net charge at your working pH, from composition and standard side-chain pKa values. Both are approximations that ignore local electrostatic environment. Attend to the sign and magnitude of exposed surface charge rather than to the third digit of the pI.

WHAT TO DO If the pI sits near your assay or formulation pH, move the buffer or introduce charge away from the interface. If net charge is strongly positive, expect non-specific binding to polyanions and cell surfaces, and test for it rather than assuming the binding you see is your target.

Low humanness and immunogenicity risk

WHAT IT IS How far the sequence resembles what a human repertoire would produce, used as a proxy for anti-drug antibody risk. A de novo binder is a sequence no human germline generated.

WHY DESIGNS GET IT Inherently. There is no human-likeness term in a structure-based generator, and for a fully de novo scaffold there is no germline to be close to. A humanness score on a designed helical bundle is being asked a question outside its training distribution.

SEQUENCE SIGNAL Repertoire-based scores are the current standard. BioPhi's OASis computes humanness from 9-mer peptide search against the Observed Antibody Space repertoire and, in the reporting paper, separated human from non-human sequences with high accuracy and correlated with clinical immunogenicity (Prihoda et al., 2022). Germline identity against a reference V-gene set is the blunter alternative, worth only as much as that set is large.

WHAT TO DO Decide what the molecule is for before spending anything here. For a research reagent, a diagnostic or an affinity tool, low humanness is not a defect. For a therapeutic ambition, record the score early because it is cheap, and understand that fixing it means humanisation or grafting onto a human framework rather than a substitution. Do not read a low score on a de novo minibinder as a prediction of immunogenicity: it was calibrated on antibody variable domains.

Polyreactivity and non-specific binding

WHAT IT IS A tendency to bind many unrelated targets. In our view the most expensive of these liabilities, because it does not look like a failure. It looks like a hit, and it survives selection.

WHY DESIGNS GET IT Generators are optimised to produce a surface that binds. Nothing in that objective asks for a surface that binds only the intended target, and a broadly sticky paratope satisfies it efficiently, typically through exposed hydrophobic or strongly positively charged patches. Selection then enriches it, because a polyreactive binder binds your target too.

SEQUENCE SIGNAL Composition and motif features rather than a single pattern: enrichment of arginine and of aromatic residues in the binding loops, and exposed hydrophobic patches. Harvey et al. (2022) built sequence-based classifiers of antibody fragment polyreactivity, showing the property is predictable from sequence for that molecule class. On a de novo scaffold such a model is a flag, not a calibrated prediction: the classifier saw antibody fragments and your molecule is not one.

WHAT TO DO Answer this one with an experiment rather than only a filter. Put counter-screens in the first selection: an irrelevant target, a polyanion such as DNA or heparin, and a negative cell line. Detecting polyreactivity in the first assay is far cheaper than discovering it after a binder has become the lead.

Deamidation and isomerisation hotspots NG, NS, NT, DG, DS

WHAT IT IS Chemical degradation of the backbone over time. Asparagine deamidates to aspartate or isoaspartate, most readily when followed by glycine; aspartate isomerises to isoaspartate, again fastest before glycine. Both change charge or backbone geometry at one position, and both accumulate on storage.

WHY DESIGNS GET IT The dipeptides involved are common and unremarkable. Nothing in a design objective penalises them, and asparagine-glycine in particular is a frequent loop motif because both residues are conformationally free.

SEQUENCE SIGNAL A dipeptide scan, which makes this the easiest class to detect and the easiest to over-report. Position and environment separate a real hotspot from a listed dipeptide. A buried, rigid asparagine-glycine is largely inert; the same pair in an exposed paratope loop is a functional liability.

WHAT TO DO Rank by position first, then consider substituting. Asparagine to glutamine, or the following glycine to alanine, removes the hotspot, and both need a re-fold check because loop geometry is often why those residues were chosen. Outside the interface, note it and carry on: a design-stage molecule does not need a shelf life.

Oxidation-prone residues exposed M, W

WHAT IT IS Exposed methionine and tryptophan oxidise, and an oxidised residue in the paratope can cost function. The same position logic as deamidation applies.

WHY DESIGNS GET IT Both are ordinary packing and interface residues, and no design objective prices their chemical reactivity. The aggregation entry applies here too: a generator that buries hydrophobic surface to drive affinity puts large hydrophobes at the interface, and the unbound state exposes them.

SEQUENCE SIGNAL A single-residue scan, and the least specific signal here: it says nothing about whether a given methionine or tryptophan is reachable by solvent. The flag means something only once exposure and position are attached. One in the core is inert; the same residue on the paratope is not.

WHAT TO DO Rank by exposure and by distance to the binding surface before substituting. Methionine to leucine and tryptophan to phenylalanine are the usual first moves, and both need a re-fold check because a large hydrophobe at an interface is usually there for a reason. Away from the interface this is a handling question rather than a sequence one: record the site and control it with buffer and storage.

A caveat on all eight. Every signal above is a sequence pattern, and a pattern tells you a liability is possible, not that it is real. Three things convert one into a finding, none of them in the sequence: exposure, structural context, and whether the site touches the binding surface. That is the next section.

Where sequence alone runs out

Sequence-level triage is fast, needs nothing you do not already have, and catches the liabilities that most often waste a synthesis order. It is also the weaker half of the analysis, and it is worth being precise about what it cannot see.

The three things a sequence scan cannot know

- **Exposure.** A hydrophobic window is a liability if it faces solvent and a feature if it is the core. A sequence scan cannot tell the difference, so it over-calls aggregation propensity on every well-packed design.
- **Spatial clustering.** Aggregation and non-specific binding are driven by contiguous surface patches, not residue counts. Five hydrophobic residues spread across the surface behave nothing like the same five packed into one patch, and a composition metric scores them the same.
- **Non-local proximity.** Residues far apart in sequence can be adjacent in space. A patch assembled from three separate loops is invisible to any window, and a cysteine pair that looks plausible by count may be twenty angstroms apart.

What structure-based analysis adds

The established structural approach is spatial aggregation propensity. SAP was introduced by Chennamsetty et al. (2009), who used full-antibody atomistic molecular dynamics to identify aggregation-prone regions and then mutated those regions to produce variants with significantly enhanced stability relative to wild type. The companion metric for viscosity is the spatial charge map. Both score a surface patch rather than a sequence window, which is exactly what a sequence scan is missing.

Both also depend on molecular dynamics, which is computationally demanding (Kalejaye et al., 2024). That cost is why these methods are rare in a design triage loop: running MD on a hypothetical two hundred candidates to decide which twenty to order inverts the economics that made generative design attractive in the first place.

Sequence-based surrogates are the practical answer. DeepSP is a convolutional network trained on 20,530 antibody sequences that predicts SAP and spatial charge map scores across antibody variable-region domains directly from sequence, with no MD run. Across 30 predicted properties the linear correlation between DeepSP scores and MD-derived scores ranged from 0.76 to 0.96, averaging 0.87 (Kalejaye et al., 2024). That is a useful ranking signal at negligible cost.

READ THE TRAINING SET BEFORE YOU TRUST THE SURROGATE

DeepSP was trained on antibody sequences and predicts properties of antibody variable-region domains. A de novo designed helical minibinder is not an antibody variable domain. Applying an antibody-trained surrogate to it is extrapolation, and the reported correlations do not carry over. The same caution applies to every antibody-trained developability model in this guide, and it is the subject of section six.

The pragmatic middle ground

You usually do not have to choose. A generative campaign produces predicted structures as a by-product, because the fold-prediction step that gave you an interface confidence score also gave you coordinates. That is enough to compute what a sequence scan is missing, with no MD: per-residue solvent accessibility, used to filter every sequence-level flag down to the exposed ones; surface patch clustering of exposed hydrophobic and charged residues, to find contiguous patches rather than counting residues; and interface mapping, marking which residues contact the target.

What you give up by staying purely sequence-level is not subtle: accurate triage of the aggregation and hydrophobicity classes, and the interface or non-interface label that decides whether a deamidation hotspot or an exposed patch is a real problem. Sequence-level screening is still the correct first pass, because it is the pass you will actually run on every design. Treat it as a filter that narrows two hundred candidates to a shortlist, then use the structures you already have on the shortlist.

A triage workflow before synthesis

Run this on the design table, before the DNA order, not after expression fails. It is a sorting step, not a gate. The output is a ranked order list with known liabilities attached, which is a more useful artefact than a pass or fail column.

- 1 **Freeze the candidate set and its predicted structures together**
One row per design, carrying sequence, predicted structure and whatever interface confidence metric your pipeline produced, joined by a stable identifier. Every later step needs both halves, and the commonest practical failure is a sequence table that has drifted out of step with the coordinate files.
- 2 **Run the deterministic scans on every design**
Cysteine parity, N-linked sequons, deamidation and isomerisation dipeptides, theoretical pI and net charge, windowed hydrophobicity, aggregation propensity. Cheap, unambiguous, no structure needed. Record every hit with its position. Do not filter yet.
- 3 **Annotate each hit with exposure and interface status**
Using the predicted structure, label every flagged position buried or exposed, and interface or non-interface. This step converts a long list of patterns into a short list of findings. Skipping it is why sequence-only screens get a reputation for crying wolf.
- 4 **Sort each finding into fix, discard, or carry with a watch**
Fix and re-run when the liability is away from the interface and removable by a conservative substitution: an unpaired cysteine, a non-interface sequon, an exposed hydrophobic patch on the back face. One substitution, re-fold, confirm the interface is unchanged, and the design rejoins the pool. This is the majority case and it costs compute.
Discard when the fix would damage the binder, when a design carries several independent high-priority liabilities at once, or when the same liability survives two fix attempts. With hundreds of candidates, discarding is cheap and a stubborn design is rarely worth the rescue.
Carry with a watch when the liability sits in the paratope and cannot be removed without trading away the binding you designed for. Order it deliberately, write down what you expect to go wrong, and design the assay to test that: an aggregation control, a counter-screen, a mass check for glycoform heterogeneity. A watched liability is a hypothesis, not an oversight.
- 5 **Order a diverse panel, not the top of the list**
Do not order the clean designs only. A panel with no liabilities can also be a panel with no binders, and a screen tuned to reject will preferentially reject the hydrophobic, high-burial interfaces that bind best. Order across the liability classes, keep the watched designs in, and use the triage table to interpret the results rather than to select the inputs.
- 6 **Feed the outcome back into the design constraints**
The cheapest developability fix is the one you never have to make. If a generator keeps producing sequons or unpaired cysteines, constrain the design alphabet or add a post-generation filter at that stage of the pipeline. One campaign's triage results tell you which constraint is worth adding, and that is what keeps this from becoming a permanent manual tax.

WHY THIS ORDER

Scan everything before filtering anything, and annotate before deciding. The temptation is to filter early because the flag lists are long, but a flag filtered before it carries exposure and interface status is a flag you cannot act on correctly. The expensive mistake is not a missed liability. It is discarding a good binder over a buried pattern that was never going to matter.

Using published thresholds honestly

This is the part of developability practice most often applied wrongly. A published threshold is a population statistic about a class of molecules. If your molecule is not in that class, it is a flag and not a verdict.

What the thresholds actually are

The reference dataset is Jain et al. (2017). Sequences from 137 antibodies in advanced clinical stages, 48 of them approved for therapeutic use, were built as isotype-matched IgG1 molecules, expressed in mammalian cells, and run through a dozen biophysical property assays. The boundaries quoted downstream are those measured distributions: where clinical-stage molecules actually sat on each assay, offered by the authors as practical guidelines for candidates that come later. Read that for what it is. A boundary of that kind describes the spread of a population that survived development. It is not a mechanistic limit on what a protein can be.

The computational guidelines built downstream are just as careful. Raybould et al. (2019) define five metrics implicated in poor developability: total CDR length, the extent and magnitude of surface hydrophobicity, positive and negative charge in the CDRs, and asymmetry in the net heavy- and light-chain surface charges. Their abstract states that the guideline cutoffs were derived from the values seen in clinical-stage therapeutics, and that a flagging system is proposed to identify nonconforming candidates. A flagging system. That is the word most downstream users drop.

Four ways the inference breaks for a de novo binder

- **Class membership.** The distribution is over clinical-stage IgG1 antibodies. A designed minibinder is not a sample from that population, so reporting its percentile within that distribution performs arithmetic on an inference that was never licensed. Several metrics are not even defined on it: total CDR length and heavy-versus-light chain charge asymmetry both require CDRs and two chains.
- **Survivorship.** The set is molecules that reached advanced clinical stages. Molecules with the same properties that failed for unrelated reasons are absent, and molecules with unusual properties that would have succeeded were never made. The boundary describes what the industry did, not what is achievable.
- **Stage.** The measured properties concern high-concentration formulation, storage and manufacture. Your question is whether to spend synthesis budget. A threshold set to protect a commercial formulation is the wrong stringency for that decision, in both directions.
- **Metric provenance.** Thresholds are defined against a specific assay or computed quantity. Your number came from a different implementation, scale or window length, or from a sequence surrogate rather than the structural quantity the threshold was set on. Comparing them compares two different measurements.

How to use them instead

- **Rank within your own set, not against theirs.** The defensible use of any developability metric on designed binders is relative: which of my designs is most hydrophobic, most charged, most liability-dense. Same molecule class, same implementation, and it is the comparison you need to order a panel.
- **Use published cutoffs to set attention, not to reject.** A design outside the clinical-antibody range is a design to look at, and the looking produces the finding. Auto-rejecting discards binders for failing to resemble a class they are not in.
- **Report the provenance beside the number.** Which dataset, paper, implementation and parameters. A threshold whose origin nobody remembers becomes a rule nobody can question.
- **Build your own distribution as you go.** Every campaign adds designs with known expression and aggregation outcomes. A threshold calibrated on your own scaffolds, host and assays is worth more than any published number, because it came from your own molecules.
- **Do not roll flags into a score you then average.** A composite discards the position and class information that made the flags actionable. Keep the flags visible.

A published threshold answers one question: is this molecule unusual for a clinical-stage antibody? At design stage you are asking a different one, which of your designs to spend money on. The first is only evidence toward the second.

Open-source tooling and licences

Enough of this is open source to build a working triage pass without buying anything. Licence terms decide whether you can. Every SPDX identifier below was read from the GitHub licence API for that repository on 30 September 2026; each description comes from that repository's README or from the paper cited beside it.

TOOL	LICENCE (SPDX)	WHAT IT DOES
oxpig/ANARCI	BSD-3-Clause	Antibody numbering and antigen receptor classification. The standard route to IMGT, Kabat or Chothia numbering, which is what makes CDR-aware liability positions exact rather than approximate. Needs an antibody-like input.
oxpig/ANARCIi	BSD-3-Clause	A language model suite for numbering antigen receptor sequences, published under the same GitHub organisation as ANARCI. Listed here as a second option to reach for when ANARCI will not number a sequence.
prihoda/AbNumber	MIT	Convenience Python APIs over ANARCI numbering. Removes the boilerplate if you are building a pipeline rather than running one command.
Merck/BioPhi	MIT	Antibody humanisation (Sapiens) and humanness evaluation (OASis). The reference open implementation for repertoire-based humanness, built on the Observed Antibody Space (Prihoda et al., 2022).
Lailabcode/DeepSP	MIT	Sequence-only CNN surrogate producing 30 spatial properties, including spatial aggregation propensity and spatial charge map, with no MD run. Trained on 20,530 antibody sequences (Kalejaye et al., 2024).
Lailabcode/AbDev	MIT	The prediction layer above the DeepSP descriptor set, from the same group. Its README describes models predicting twelve biophysical properties of monoclonal antibodies from variable region sequences. Its stablemates DeepViscosity (MIT) and DeepSCM (NOASSERTION, so read the repository's own licence text) predict high-concentration viscosity, which is a formulation question rather than a design-stage one.
debbiemarkslab/ nanobody-polyreactivity	MIT	Sequence-based polyreactivity models trained on a naive synthetic camelid nanobody library (Harvey et al., 2022). The closest of these to a single-domain scaffold.
regeneron-mpds/propermab	Not open	Do not use commercially. The GitHub licence API returns NOASSERTION. The repository's LICENSE.md is headed "License for Non-Commercial Use of PROPERMAB code", defines "You" as an academic institution or academically employed full-time personnel only, grants rights for academic research purposes only, and prohibits use in any commercial online service. Read the file yourself before relying on this summary.
platforma-open/ antibody-sequence-liabilities	None	No licence at all. The GitHub licence API returns 404 and the repository root contains no LICENSE file. With no licence granted, default copyright applies and you have no permission to use, copy or modify it. The liability motif classes it covers are documented and easy to reimplement, so do that instead.

Building a pass from these

Write the deterministic scans yourself: a sequon regex, a cysteine parity check and a dipeptide scan are a few dozen lines and you want to own the definitions. Add the tools above for what each covers, and compute solvent accessibility from your predicted structures. That annotation step is what makes the rest trustworthy.

Two cautions that are not about licences. BioPhi, DeepSP, AbDev and nanobody-polyreactivity were each developed on antibody or nanobody sequences, as the rows above record. A de novo designed binder is neither, so section six applies to all four. And check the licence yourself: licences change, and the two stops in this table are what gets missed when a dependency arrives through someone else's pipeline.

Run your own sequence

Developability Scout is our free sequence-level screen. No account, no install. Paste one sequence, get a scored profile back. What follows was read from the tool's source on its main branch, not from its own documentation.

`tools.ranomics.com/developability`

What it computes

Five independent dimensions, combined into a single 0 to 1 composite with equal weight on each: humanness, sequence liabilities, charge, hydrophobicity and aggregation. It also returns per-residue annotations, so you can see where a flag sits.

- **Liabilities.** Deamidation and isomerisation dipeptides, the DP Asp-Pro fragmentation dipeptide, N-linked glycosylation sequons by the regex `N[^P][ST]`, the RGD integrin-binding tripeptide, oxidation-prone residues, and a free-cysteine check that flags an odd total cysteine count.
- **Hydrophobicity.** Kyte-Doolittle hydropathy averaged over an 11-residue window, with patches called above a fixed threshold.
- **Aggregation.** An Aggrescan-style per-residue propensity, smoothed over 5 residues, with aggregation-prone regions required to be at least 5 residues long and to clear a minimum mean.
- **Charge.** Theoretical pI and net charge at pH 7.4, scored as two legs and averaged: pI against a 6.0 to 9.0 window, net charge against -5 to +20, each decaying linearly outside it.
- **Humanness.** 7-mer overlap against an internal human germline V-region reference set.

What it does not do

- **One sequence per run.** The input handler strips every FASTA header line and concatenates what remains into a single sequence. Pasting a multi-record FASTA therefore does not screen a panel: it fuses your records into one chimeric sequence and scores that. Submit one sequence at a time. It accepts 10 to 2000 residues and the 20 standard amino acids only.
- **No export.** There is no CSV, no PDF, no downloadable report and no heatmap export anywhere in the tool. Results are read in the browser, so copy what you need.
- **Sequence-level only.** No structure is used, so no solvent accessibility, no surface patch clustering and no interface mapping. Everything in section four about what a sequence scan cannot see applies to this tool.
- **CDR positions are guessed from length, not read from your sequence.** CDR1 and CDR2 begin at hardcoded Kabat-like positions with only their ends clipped to your length; on the heavy path CDR3 is placed at length minus 25 through length minus 10. There is no IMGT numbering and no ANARCI call in the running code, so for a de novo binder that is not a canonical V-domain the in-CDR flag is a guess about where an antibody would have kept its loops.
- **Humanness uses a small internal reference, not a repertoire.** The germline table holds 11 V-region sequences: 5 IGHV, 3 IGKV, 3 IGLV. There is no Observed Antibody Space lookup. For a real humanness assessment use BioPhi's OASis, which searches a repertoire database.
- **Choosing scFv or Other silently applies light-chain CDR spans.** The liability scanner treats only VH and VHH as heavy, so VL, VK, scFv and Other all fall through to the light-chain positions 24 to 34, 50 to 56 and 89 to 97. Humanness splits those four differently again: VK is scored against kappa germlines only, VL against lambda only, and scFv and Other against the pooled set of all eleven. For anything that is not a plain VH or VHH, read the raw liability positions and ignore the in-CDR flag.

Used as a first-pass triage on one sequence at a time it is genuinely useful. Used as a panel screen or as a verdict it will mislead you, in the ways listed above.

Where Ranomics fits

IF YOU HAVE A TARGET BUT NO DESIGNS YET

Epitope Scout, also free, ranks candidate epitopes on a target before any design run, at ranomics.com/technology/epitope-scout. It answers a different question from this guide, which is where to aim rather than what to screen.

We design and validate de novo binders, and developability triage is part of how we decide what to synthesise. If you want the workflow in this guide run on your target rather than assembled yourself, there are three entry points, in increasing scope.

- **Binder Pilot**, at ranomics.com/binder-pilot. The starter: a smaller design pool against one target, screened through one FACS or MACS yeast display selection round, delivering ranked binder hits with NGS readout and a technical report. Built for academic labs, seed biotech, industrial SMBs and student research groups, and soft agreements such as a mutual NDA or an MTA are fine.
- **AI Binder Sprint**, at ranomics.com/ai-binder-sprint. Our flagship, six to eight weeks: a multi-algorithm design pool, filtered through structural scoring and developability checks, and only the survivors are synthesised and screened by display.
- **Custom campaign**, at ranomics.com/biotechnology-services. Open scope: multiple targets, multiple rounds, experimental validation by yeast and mammalian display, pharma-grade work.

None of the three has published pricing. Scope drives the number, so we quote per project. Tell us the target and what you need at the end of it, and we will tell you which of the three fits at ranomics.com/ranomics-contact.

References

1. Chennamsetty N, Voynov V, Kayser V, Helk B, Trout BL. Design of therapeutic proteins with enhanced stability. *Proc Natl Acad Sci USA*. 2009;106(29):11937–11942. doi:10.1073/pnas.0904191106
2. Conchillo-Solé O, de Groot NS, Avilés FX, Vendrell J, Daura X, Ventura S. AGGRESCAN: a server for the prediction and evaluation of "hot spots" of aggregation in polypeptides. *BMC Bioinformatics*. 2007;8:65. doi:10.1186/1471-2105-8-65
3. Harvey EP, Shin JE, Skiba MA, et al. An in silico method to assess antibody fragment polyreactivity. *Nat Commun*. 2022;13(1):7554. doi:10.1038/s41467-022-35276-4
4. Jain T, Sun T, Durand S, et al. Biophysical properties of the clinical-stage antibody landscape. *Proc Natl Acad Sci USA*. 2017;114(5):944–949. doi:10.1073/pnas.1616408114
5. Kalejaye L, Wu IE, Terry T, Lai PK. DeepSP: Deep learning-based spatial properties to predict monoclonal antibody stability. *Comput Struct Biotechnol J*. 2024;23:2220–2229. doi:10.1016/j.csbj.2024.05.029
6. Khakerwala Z, Kumar A, Gaikwad SS, et al. Experimental analyses of RFdiffusion designed Miniproteins for binding to SARS-CoV-2 nucleocapsid protein. *Protein Eng Des Sel*. 2025;39:gzag004. doi:10.1093/protein/gzag004
7. Kyte J, Doolittle RF. A simple method for displaying the hydropathic character of a protein. *J Mol Biol*. 1982;157(1):105–132. doi:10.1016/0022-2836(82)90515-0
8. Pacesa M, Nickel L, Schellhaas C, et al. One-shot design of functional protein binders with BindCraft. *Nature*. 2025;646(8084):483–492. doi:10.1038/s41586-025-09429-6
9. Prihoda D, Maamary J, Waight A, et al. BioPhi: A platform for antibody design, humanization, and humanness evaluation based on natural antibody repertoires and deep learning. *MAbs*. 2022;14(1):2020203. doi:10.1080/19420862.2021.2020203
10. Raybould MIJ, Marks C, Krawczyk K, et al. Five computational developability guidelines for therapeutic antibody profiling. *Proc Natl Acad Sci USA*. 2019;116(10):4025–4030. doi:10.1073/pnas.1810576116
11. Watson JL, Juergens D, Bennett NR, et al. De novo design of protein structure and function with RFdiffusion. *Nature*. 2023;620(7976):1089–1100. doi:10.1038/s41586-023-06415-8

Ranomics · ranomics.com · Developability Assessment for De Novo Designed Binders. Tool behaviour described in section eight was read from the Developability Scout source on its main branch on 30 September 2026 and may change; the tool's own interface is authoritative. Licence identifiers in section seven were read from the GitHub licence API on the same date. Nothing here is a guarantee of any experimental outcome.