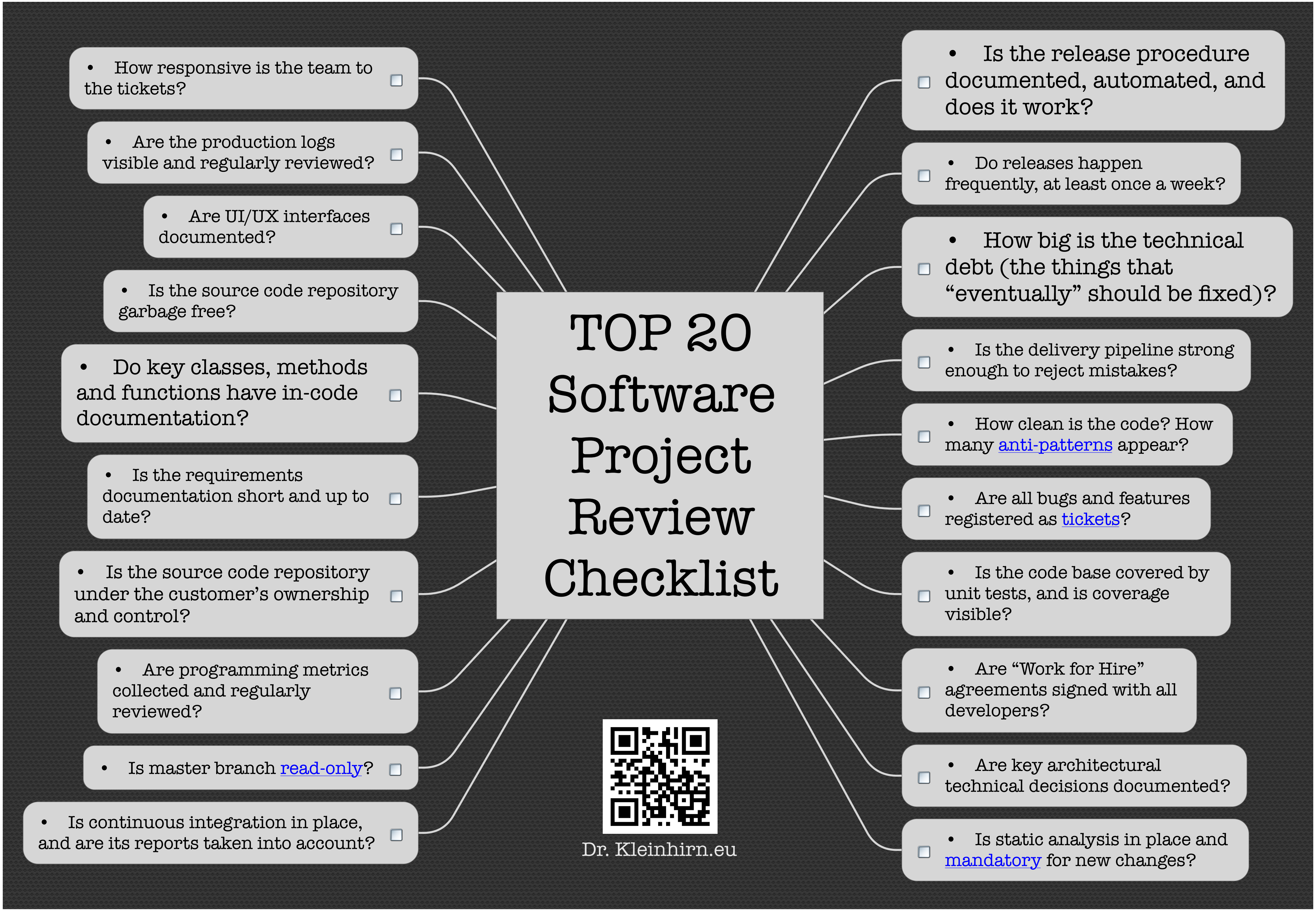

TOP 20 Software Project Review Checklist

- How responsive is the team to the tickets? □
- Are the production logs visible and regularly reviewed? □
- Are UI/UX interfaces documented? □
- Is the source code repository garbage free?
- Do key classes, methods and functions have in-code documentation? □
- Is the requirements documentation short and up to date? □
- Is the source code repository under the customer's ownership and control? □
- Are programming metrics collected and regularly reviewed? □
- Is master branch read-only? □
- Is continuous integration in place, and are its reports taken into account? □

- □ Is the release procedure documented, automated, and does it work?
- □ Do releases happen frequently, at least once a week?
- □ How big is the technical debt (the things that "eventually" should be fixed)?
- □ Is the delivery pipeline strong enough to reject mistakes?
- □ How clean is the code? How many anti-patterns appear?
- □ Are all bugs and features registered as tickets?
- □ Is the code base covered by unit tests, and is coverage visible?
- □ Are "Work for Hire" agreements signed with all developers?
- □ Are key architectural technical decisions documented?
- □ Is static analysis in place and mandatory for new changes?

Dr. Kleinhirn.eu