



Test Task for a Freelance Copywriter

IA: Software Security Issues

The reader's situation: A manager/owner of a software product start-up wants to learn experience-based tips on how to develop secure software.

Structure:

- 1.** "Editor's Note" - a special section aimed at making the reader understand from the very beginning that ScienceSoft doesn't just write for informational purposes; it is a company that provides the relevant services. This section can contain, for example, a brief overview of the article and an invitation for a reader to read our service offer with a link to the relevant service page (<https://www.scnsoft.com/services/software-development/saas>).

Example from one of our articles:

Monolith Pains Sound Too Familiar?

Our professional BAs, reverse engineers and architects can help to split it into manageable and understandable parts (web services/microservices) to boost its agility reliably and with minimum disruption to your business.

[SPLIT MY MONOLITH](#)

- 2.** Short introduction with a link to statistics.

Also, we need a link to <https://www.scnsoft.com/blog/secure-software-development-guide> with [security in software development] anchor.

- 3.** List of actual security issues (4-5) with descriptions and protection tips.

- 4.** A call-to-action in the middle of the article for a reader to contact ScienceSoft's team. It should contain a heading, main text describing how ScienceSoft's team can help and short text for a CTA button.

Example from one of our articles:

Monolith Pains Sound Too Familiar?

Our professional BAs, reverse engineers and architects can help to split it into manageable and understandable parts (web services/microservices) to boost its agility reliably and with minimum disruption to your business.

[SPLIT MY MONOLITH](#)

5. Short conclusion with a soft selling intent and a call-to-action offering to contact our team.

Example from one of our articles:

The value of measuring technical debt

I believe that measuring technical debt is the foundation of its management strategy. It will show how debt impacts the project at the current stage and help allocate reasonable time and resources on eliminating it. And to make debt estimation more precise and efficient, you need to implement automated tools and make their usage a part of your software development routine.

However, in some cases, the roots of technical debt may lie deeper than just existing bugs and complex code. This could be a poor architecture choice, weak QA strategy or lacking project management. If you want to understand the reasons behind never-ending technical debt in your project and get this problem solved or set the project right from the very beginning, you may consider [requesting our software consulting services](#).

Length	URL	Keywords	Body
600-800 words	/blog/software-security-issues	<ul style="list-style-type: none"> • Title: software security issues • H1: software security, issues • Meta-description: software security, issues 	<ul style="list-style-type: none"> • software security issues • software security (2) • application security • security issues • vulnerabilities

Competitors:

- darkreading.com/application-security/10-common-software-security-design-flaws/d/d-id/1306776
- searchsecurity.techtarget.com/feature/The-future-of-software-security-vulnerabilities
- bryansoliman.wordpress.com/2011/07/27/security-issues-in-software-development/

Submit the completed task to ScienceSoft's Chief Editor at mvolkova@scnsoft.com along with your CV and portfolio, if any.