How can an improper code base undermine a start-up?

#	Issue	Improper code base …
1	Technical debt	… leads to a never-ending cycle of bug fixes, maintenance, and updates, taking away resources that could be allocated to product development and growth
2	Complexity and maintainability	… makes it difficult for new team members to onboard, increasing the time and effort required to maintain the system
3	Security vulnerabilities	… can leave the start-up vulnerable to cyber threats, data breaches, and reputational damage
4	Scalability issues	… may not be able to scale with the growth of the business, leading to performance issues, slow response times, and decreased user satisfaction
5	Lack of testing and quality assurance	… can lead to a lack of thorough testing, resulting in bugs, errors, and inconsistencies that can negatively impact the user experience
6	Inefficient use of resources	… may waste resources on unnecessary features, complex algorithms, or redundant code, leading to decreased productivity and competitiveness
7	Difficulty in attracting and retaining talent	… can make it challenging to attract and retain top talent
8	Increased risk of failure	… can increase the risk of project failure due to technical issues, maintenance costs, or scalability problems
9	Poor user experience	… can result in a subpar user experience, leading to decreased customer satisfaction, loyalty, and ultimately, revenue loss
10	Lack of flexibility	… may not be adaptable to changing business needs, making it difficult for the start-up to pivot or respond to market shifts