1. Reward points handled by client
2. Database uses legacy authentication
3. Tokens do not expire
4. Passwords not salted in database
5. Passwords use MD5 as storage
6. No way to change password
7. user\_get returns birth date, potential security vulnerability
8. users can request multiple (valid) tokens
9. No limits on returned posts, potential loss of availability for crashing / slow
10. REST API has no (Rate) limits or quotas imposed – potential for crashing
11. API is perhaps a bit too open in certain points – for example, anyone can get a list of all posts associated with a user
12. Users could potentially create their own application to contact backend API – no ideal