### **Functional Requirements: What can users do with it?**

* The complete list of functionalities for each user type is maintained on Notion, [please visit this link ↗️](https://shubrank.notion.site/86ec14acfd724fd1a9638a6853a68678?v=5e48750a1b4c4685af503511ea843225)
* The list of functionalities identified are based on the review processes and SOPs outlined here: [Application review process, SOP & policy 2.0](https://docs.google.com/document/d/1vb3DBYgSR8M5DcGhXTmc3ZGUNgT8f4yVL0MMvOCo-tU/edit#)

### **Non-Functional & Other Requirements**

1. Webapp should be open source. Code should be maintained publicly on GitHub under the DPGA account. OS licence to be used: <to be decided>
2. Webapp should be able to handle upto one thousand simultaneous users to begin with. This traffic includes DPGA reviewers, public reviewers and applicants.
3. Webapp should be hosted on **app.digitalpublicgoods.net**
4. A copy of all applications (except for PII data), including new applications and rejected applications, should be made publicly available in a GitHub repo.
5. A copy of all applications (except for PII data) will be available at a direct public link. Example: app.digitalpublicgoods.net/application/<id>
6. Application IDs will be starting from 10001 and will increase in steps of 1 with each new application.
7. DPG registry should get updated in near real-time based on decisions made through the app.
8. All automated system-generated emails should be sent from an unmonitored email ID.
9. The application form should be publicly accessible without the need for signing up. Signing up should be required to save progress. Email verification should be required to submit an application.
10. Applications in any list should be sorted from largest to smallest score based on the following logic:  
      
    Priority application = 6 points  
    Refresher application = 3 points  
    Clarification submitted = 2 points  
      
    Sorting for applications with the same score will be on the basis who submitted the application first.
11. All activity on the webapp should be time-stamped.

### **Tech Stack**

* Backend - Codeigniter (PHP)
* Database - MySQL
* Frontend - Bootstrap 5
* APIs - REST APIs
* Hosting - Apache server with MySQL (basic hosting or AWS EC2 server)

## **How to contribute or submit feedback**

If you are a DPG applicant and have suggestions for the process, comment below so our team can look into the request and convert it into a pull request.