

Link Checker WebApp

Web application to check several backlinks grouped by project automatically

Cordoba, Argentina, 2017-09-20
ELCAN SOFTWARE



Mario O. Villarroel
Customer Care/System Design
movilla@elcansoftware.com

Juan P. Aguilar
System Development
aguilarjpb@elcansoftware.com

1 Document Version Control

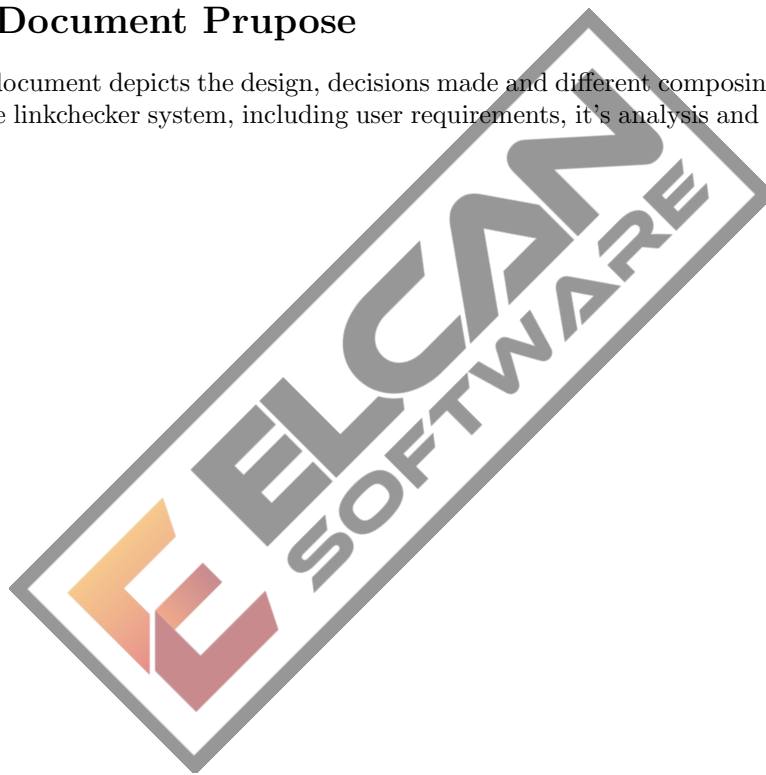
Version	Author	Reviewer	Approver	Date	Change log
1.0	Mario O. Villarroel	J.P. Aguilar	Zaitec	2017-09-20	Initial Version

2 Project distribution list

Name	Role	E-mail	Date	Signature
Mario O. Villarroel	System Engineer	movilla@elcansoftware.com	2017-09-21	-
Juan P. Aguilar	System Developer	aguilarjpb@elcansoftware.com	2017-09-21	-
Zaitec	Product Owner	zaitec@fiverr.com	2017-09-21	-

3 Document Prupose

This document depicts the design, decisions made and different composing parts for the linkchecker system, including user requirements, it's analysis and derivations.



4 System Requirements

4.1 Customer Requirements

The system will be a web interface that allows linkback checking, it should be able to handle several "projects" where the links will be listed and show its status.

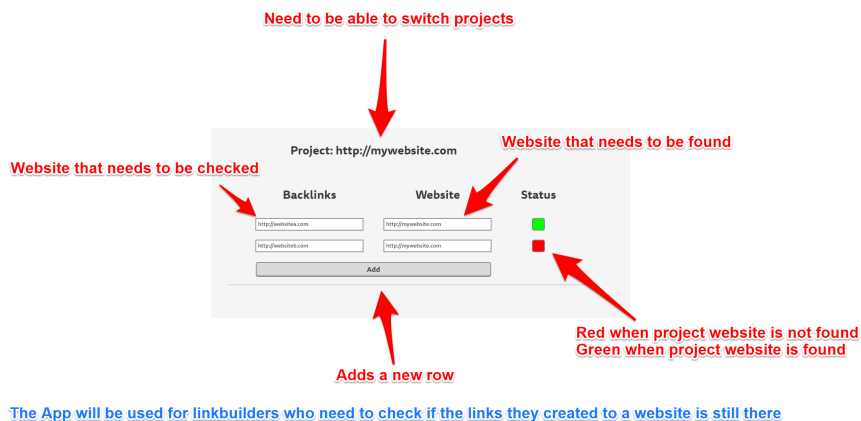


Figure 1: User provided details

4.2 Requirement Analysis and elicitation

The following table lists the user requirements from the original system request.

1. The system shall allow the unique user identification
2. The system shall ask for username and password to access it
3. The system shall identify a special user as administrator who will be able to
 - (a) Add other users to the system.
 - (b) Permanently delete objects from the system.
 - (c) recover user deleted items.
 - (d) block users from login in.
 - (e) Unblock the users blocked.
4. The system shall allow to list, create, modify and delete "projects" to group link checker items.
5. The system shall allow to list, create, modify and delete "link checker" items.
6. The system shall check the link status automatically when displaying the checker list

7. The system must be developed in PHP
8. The system could be developed using the Laravel framework.
9. The system will have an admin template in use for it's display.

5 System Design

5.1 Entity Relationship Diagram

Please refer to figure 2

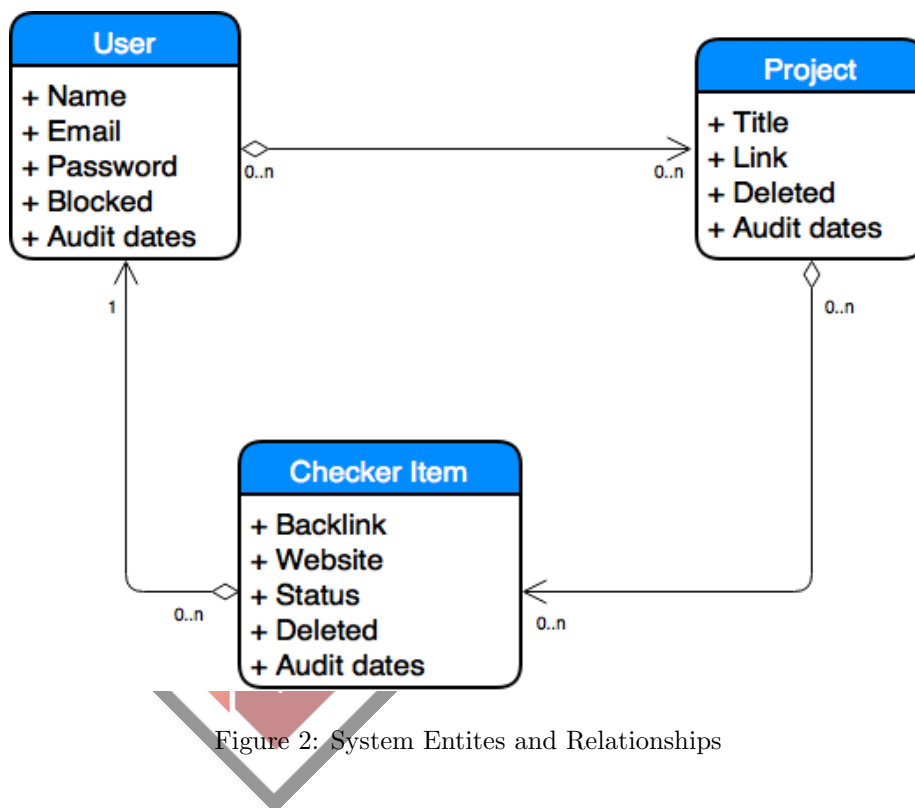


Figure 2: System Entites and Relationships

5.2 Use Cases

Please see figure 3

5.3 Design desitions

- The link checker items list will be paged
- The link checker items page will have 15 items by default, 25 as a second option and 50 as the last one.
- The Angular library will be used for the interface wherever it applies.

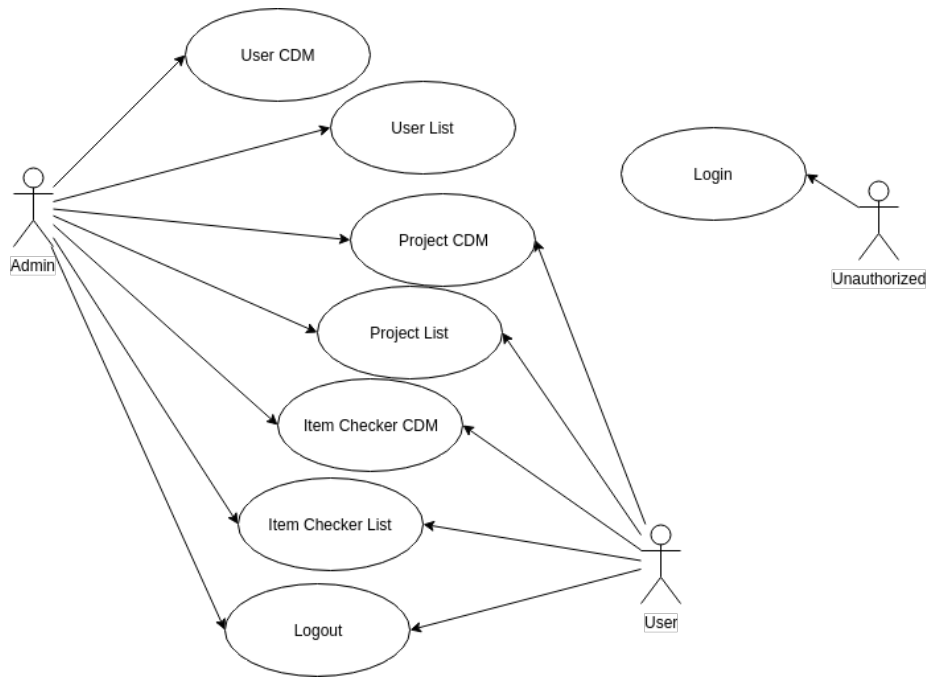


Figure 3: System use cases

5.4 Possible future features

After doing the requirements analysis, we've come to a list of possible future/extension features which we list below:

- Groups/companies.
- Distinguished user as manager for a group/company.
- New low user level with less permission and under a manager's group/company.
- Managers are the only users able to create projects and assign users to it.
- low users can only create links on their assigned project.

This features can be accomodated in a new project if needed.