Projectdefinition

1. Software Engineering Class:

* A project for the lectures "Software Engineering" with Mrs. Key Berkling will be car­ried out. During this project a software application is to be developed. At the end of the project period, this will serve as an evaluation basis for the lecture.

1. Product idea:

* The application described here is called BFFL and is a web application that can be ac­cessed via a web browser. For the development of the application the framework An­gular is used. Java will also be used for a rest service.
* The application of this team is used to "shorten" URLs. This means that any URL can be entered with sufficient length and the user is shown a URL that has fewer cha­rac­ters and still leads to the same destination.
* These shortened URLs can be associated with an account or group of accounts. This al­lows users to view and change the stored URLs. This is especially interesting if the same URL is used 100 times, for example, and you want to change the destination of these URLs after a certain time. This way you do not have to change 100 URLs but on­ly the target URL in BFFL.
* In addition, it is possible to pull data from the clicks on URLs and use them yourself. For example, one could calculate the frequency of use, time of day dependencies and much more. This is rarely possible with other URL shorteners, as the data is not made available to the user of the URL shortener. BFFL guarantees 100% secure and full access to YOUR data sets.
* Since BFFL is intended to be hosted as an application on the company's own servers, cus­tomers of BFFL avoid another risk/problem. In the past, it has often happened that the URLs of shortened services were deactivated and customers' websites were no longer accessible or other important functions could no longer be used.

1. Customer: BFFL-Team
2. Project-Team: Felix Hirschel, Lars Hudalla, Bastian Schäfer, Fabian Schwickert
3. Requirements analysis:

Four essential components are required for the web application:

1. A frontend to create, edit and view short URLs. Furthermore, it should be possible to display complete data from URL usage.
2. A residual service in the backend, which is addressed with a click on a shortened URL and redirects to the correct destination address.
3. A DBMS in which the data records for URLs and users are stored.
4. An access management system that identifies users when they log in to the front end. There must also be two roles that distinguish between users and admins.

**The utilization of the application should be as follows:**

1. A user opens BFFL in the browser. Now a web page should be displayed which asks for a login or registration.
2. The User must first register with his e-mail address, a password and his name. Op­tio­nally a verification mail can be sent.
3. With a click on Submit, the database is checked to see whether such a user already exists. If this is the case, you should be redirected to the registration page. Other­wise an entry in the database is created and redirected to the login page.
4. The user enters his credentials and presses Login. The database checks whether the user has entered valid and correct credentials. If not, the user is redirected to the login page. Otherwise he will be redirected to the main page of BFFL.
5. Now it is checked whether the registered user already belongs to a group (data­base). If this is not the case he has to create a group or join a group to use the app­li­cation. Because the URLs are always assigned to a group.
6. A pop-up appears, allowing you to create a group or specify an ID to join the as­so­ci­ated group. After selecting one of the two options and entering the night data, the da­tabase checks whether it is possible to create the group or join the selected group. If an error occurs, the process must be repeated. If successful the app­li­ca­tion will be unlocked.
7. Now a URL must be generated. To do this, fill in the form on the main page in the area that is overwritten. With a Submit the data is checked and then a data record is created or an error is thrown.
8. The user is now shown a pop up which shows the new entry again and gives the op­portunity to test the new URL.
9. With a click on the URL, a request is now made to the rest of the service in the back­end. This holds various data from the HTTP request. It then searches the da­ta­base for a suitable entry for the URL called up. In addition, new entries are created which concern the use of the specific link. Finally, the service forwards the HTTP request to the appropriate target URL.
10. On the Main Page the user can now view a list of all the links in his group. With a click on an entry he can change or delete it, provided he has the rights to do so. With the necessary rights a pop up opens, which shows the whole entry as a form. Here you can delete the entry or enter new values for all fields by pressing the but­ton.
11. If the logged-in user is a member of more than one group, he can switch between the projects using the drop-down menu in the Top Bar. In addition, in a project where he is admin, he can access the data analysis page in the Top Bar.
12. Project goal:

* The aim of the project is to produce a first executable version of a URL shortener that implements essential functions.
* Therefore the functionality for data analysis should not be implemented first. For the time being, the focus is on the URL shortening and the benefit itself.

1. Approach:

* **First**, sufficient planning must be carried out.
* **Secondly**, the technical infrastructure must be set up. This includes a web server and a DBMS. It must also be clarified which tools, platforms etc. are used.
* The **third** step is the processing of tasks related to the development of the app­li­ca­tion. This means coding and creating DBMS instances, access management in­stan­ces, etc.

1. Scope:

* The project period ends in June 2021. However, there is not yet an exact date for the pro­duct handover.