Project documentation - 2018/2019 - V1.0

Common functionality for all projects:

1. **Unregistered user -** user that has no account on the system. To become a member, it must register on the system where he must enter personal information about himself, username, password and password repeated. User is considered registered when he activates his account over a link got over e-mail. Activation link is valid for 7 hours. Every unregistered user when he first visits the webpage must accept terms of usage which are related to recording data into cookies and this is saved into cookie (which is valid for 2 days) so it is not necessary to accept this every time.

2. **Registered user –** user which has an activated user account. Login consists of entering a username and password. If the user enters (in a row) wrong login data, the account is locked; in this case unlocking can only be done by administrator. If the login is successful a session is created which is valid unit a time specified by administrator or until logout. Registered user has all rights as the unregistered user. Cookies have information about terms of usage about cookies which last one moth for the date of registration for registered user, after that it needs to be renewed.

3. **Moderator –** has all rights as registered user.

4. **All roles –** on every part of the project where creation of objects (e.g. Categories, users, etc.) it must be also possible to update and view the same objects.

5. **Administrator –** has all rights as all previously defined roles, beside that it has the possibility to create roles in the system and it has CRUD rights (create, update, delete) on all data. This must be manual solution for every table. Administrator has the possibility to view the system log (look specification point 9) as also the possibility to search/filter the log based on: date (time interval), type of log entry and user. Administrator can also view the list of locked user accounts and unlock the user accounts. He can reset the terms of usage where the user need again to accept them (point 1) and additionally configure the application (e.g. Cookie duration, pagination, etc.).

6. **Pagination –** every data listing with more than 7 rows must have pagination. It is a plus to have the possibility to configure number of displayed rows on one page by the administrator.

7. **Statistics –** All statistics must have the print preview with a button for print (which opens print dialog in browser) and the PDF generate button (for extra pints). All statistics must have the possibility to sort data based on presented columns (at least for 2 columns). **It is mandatory to view statistical data also in graph form (at least one graph types of your own choosing**. I is forbidden to use done solutions (PHP and JavaScript), you must your own implementation using JavaScript and Canvas and you can use AJAX if you want.

8. **Relative paths –** in the whole project relative paths must be used.

9. **System log –** every request/action must be written into system database log (who, when, what was done, etc.). It is necessary to record all basic information about systems actions into log (type login/logout (user, time), queries on database (user, query, and time), type of action (user, time, and action)). Administrator can view the log report in a wanted interval (chronological view of all or chosen users, frequency of work for some users, etc.)

10. **Search and sort –** every table view must have the possibility to search and sort (at least for two columns) data.

General specification

11. **Authentication –** must be done with own method over your database. It is necessary to validate data on client and server side. You must ensure security and protect the web application form automated registering of users. It is wanted to record information about last successful login (only username). More detail about login are written in point 2 of this specification.

12. **Passwords –** all passwords need to be saved in database in two columns. First is readable format and second in encrypted form using SHA1 or SHA256 hash algorithm and usage of salt. Readable form is only used for practical reasons in case you forget your passwords.

13. **HTTPS –** login must be implemented over secure line meaning over HTTPS protocol. Rest of the project can be done over HTTP.

14. **XSS and SQL injection protection –** for extra points crate basic protection against XSS attack using filter_input on input and htmlspecialchars on output and protection against SQL injection using prepare statements.

15. **User interface –** must be implemented using AJAX which will take data from server (we recommend using XML or JSON, while usage of HTML will get you less points). jQuery can be used for specific parts of project; more usage of jQuery gets you more points. For data views it is necessary to implement paging (for more detail look point 6 of this document). It is a plus if you implement personalization and help/ease of use for the user (for example by using multiple CSS).

16. **Data in the database –** main storage of data like (users, family trees, cars, parts, equipment etc.) must contain at least 20 different rows. Other tables must contain more than 30 different rows.

17. **Virtual time –** all work of the application is based on virtual time which is calculated based on real time on the server and a specific time shift is added. Unit of shift is 1 hour. Administrator of the application is the only one that can change the time shift, for example he can move the time 7 days forward by setting the shift to 148hours or 7 days in the past by setting -148hours. Once the administrator put in the time shift

all activities form that point use that virtual time. It is suggested that a special function is made to return the virtual time and which is used in all time activities. Steps to implement the virtual time are the following:

- Setting up the time shift by entering number of hours in the following web address: http://barka.foi.hr/WebDiP/pomak_vremena/vrijeme.html.
- Getting the time shift to your application for one of the following links: http://barka.foi.hr/WebDiP/pomak_vremena/pomak.php?format=xml or http://barka.foi.hr/WebDiP/pomak_vremena/pomak.php?format=json. These are xml or json files of simple structure. Your application needs to use this XML or JSON format to get the time shift and then save the sift into your application somewhere in the database or configuration file. This is only done when admin user requests it.
- Every time a date time activity is done it is based on the virtual time shift that is stored in your database and the real time on the server. The set-up time shift is used in the application until the admin user changes it.

18. **Installation –** solution bust be installed on **barka.foi.hr**. Access must be denied to other students.

19. **Data –** are stored in MySQL database under the name WebDiP2018xnn (nn is the number of student 01 <= nn <= 300). For all students databases are created and privileges are assigned to them to enable work on the database. Students got the necessary information over email from the teacher.

20. **Directory –** on which the solution is stored is WebDiP2018xnn (nn is the student number 01 <= nn <= 300) inside the directory /var/www/WebDiP/2018_projekti. Students will get this information together with the database information. In this directory only data and scripts related to the project solution must be present.

21. **Directory "private"** – this directory is part of the application and must be protected with .htaccess file whereby in this file must use a user with the same username and password that is used to access the database (look at point 16). Script private/users.php must list all users and their passwords in readable form. Access is NOT restricted based on the application users.

22. **Documentation –** File name must be documentation.html which is accessed from index page. Documentation must contain:

    a. Description of the project assignment
    b. Description of the project solution
    c. Important parts of the solution (ERA model)
    d. List and description of all scripts, site map and navigation diagrams
    e. List and description of used technologies and tools
    f. List and description of external modules/packages/libraries and their usage in the scripts and similar.

23. **About the author –** File name must be about_author.html which accessed from the index page. Page must have image (like in user ID), name, surname, user id number and mail (with link to open a mail to send). Rest contains other information as you wish.

Defence of the project

- Apply for one of the given times for project defense

- Install the solution to **barka.foi.hr** according to previous request in this document

- Bring with you a laptop on which the project will be presented. Laptop must have at least two browsers (Chrome, IE, Firefox, Opera) because the presentation will be done in parallel on both browsers. Laptop must have installed necessary tools to access Eduroam wireless network and **ensure to work with your credentials**.

- **Fill out whole** project form, print it, sign it and bring to your project defence. Students that do not fill out the form until the beginning of the defence will lose some points that they gather.

- **Fill out whole** form for evaluation (self-evaluation) of the project, print it, sign it and bring it to the project defence. Students that do not fill out the form until the beginning of the defence will lose some points that they gather.

- The whole project solution and documentation (directory structure) including the database export under the name WebDiP2018xnn.sql (nn is number of the student 01 <= nn <= 300) must be saved as zip file and be brought to the defence. Student will submit the prepared zip file on Moodle system after successful project defence.

- Prepare the presentation of the solution where all important parts will be presented. Teacher will ask question and guide the presentation to some extent during the presentation if it will be necessary.

- Come to the presentation 10 minutes early before the beginning in front of the professors room and make the laptop ready to work.

- After defence while still in the presentation room students must forbid access to their project source code on the server and MUST NOT unlock it at any lather time!