Criterion E: Evaluation of a product

Evaluation of the success criteria was made according to the client’s feedback.

Full definitions of the success criteria are listed in *Criterion A*.

| Criteria | Is the criteria satisfied? | How is it made? |
| --- | --- | --- |
| 1 | Yes | *XML* language and *Android Studio* platform allowed to create effective, yet simple interface with no distracting details which would be unnecessary. |
| 2 | Yes | Real-time database *Firebase* was connected to the application, which allowed an optimised data transfer between the application and the database. |
| 3 | Yes | Bubble sort was used to effectively sort events by date in the ascending order. |
| 4 a | Yes | OOP provides the creation of separate events which can have the same date. When object is stored to the database, *Firebase* generate unique id for the object thus several events of the same date can be created. |
| 4b | Yes | Appropriate buttons were created which allow user to delete the event and remove it from the database. |
| 5a | Yes | A special *Boolean* is created, which determines whether the user is a Moderator. Appropriate checkboxes are created which allow Moderator to turn volunteer into a moderator. |
| 5 a | Yes | A special *Boolean* is created, which determines whether the user is verified. Appropriate checkboxes are created which allow Moderator to verify volunteer. |
| 6 a | Yes | Methods are implemented which check whether all the necessary entries during the registration are filled properly. |
| 6 b | Yes | Method is implemented which checks whether the string of the password has more than 6 characters |
| 7 | Yes | A special window was created which allows user to retrieve the user’s data from the database and edit it. |

**Further development of the program**

During several meetings with **the client**, the future development of the program was discussed.

Firstly, the connection between the application and the local Philharmonic should be made. The events could be implemented from the web-page, thus the Moderators could create less events by themselves.

Secondly, it would be convenient for volunteers, if the app provided notifications for those users who are going to a particular event an hour before the start of the event.

Moreover, automatic deletion system could be provided - the day after the event happened it would be automatically deleted, so it would not confuse users as well as use the storage of the database.

Also, the client suggested that the application could count the number of events a particular volunteer has attended. This would provide moderators with information which volunteers are more active and thus might even give additional rewards.

Furthermore, **client** suggested making the app more suitable for other organisations which needs volunteers. Therefore, during the registration process user should be able to choose the organisation they are willing to volunteer in. Also, the list of the organisations which would want to use the application should be implemented, as well as the User and Event objects modified, which would allow them to be put in the separate section of each organisation.

Word count: 484