Presentation App

MIS 2023 Summer Semester

Project Description

INTRODUCTION

This project is to be executed for the evaluation of the course 'Mobile Information Systems' during the summer semester of 2023.

The project is to be carried out in groups of maximum 3 students. Each group designs, implements and tests a mobile application in the context described in this document. The result is presented to the other groups during the last course session in room G101. During the presentation a demonstration is to be done on a physical device – not an emulator.

Context

The target group for this mobile app consists of presenters and their audience. Every user of the app can be a presenter or part of the audience. To become a presenter, the user initiates a presentation which other users can follow. By following a presentation, the user is part of the audience.

A presentation consists of slides that are shown in the app to the audience. When the presenter proceeds or navigates back to a slide, the screen of the audience is updated accordingly.

PROJECT DESCRIPTION

For this project, you develop a working prototype of the mobile application using Google Flutter, either runnable on an Android device or an iOS device. You may choose the target device (smart phone, tablet) and layout (portrait, landscape) for the application based on the functionalities you will implement and the usability in the real-life environment the application is to be used.

You also perform a well-documented user test.

Presenters

A user acts as a presenter when he/she initiates a presentation (tapping e.g., 'New presentation'). The first step for the presentation is to choose the slide deck that will be shown in the presentation. It is up to you how you implement this functionality (selecting a file, selecting from a database list…). After the slide deck is selected, the presentation starts and the first slide is shown.

The presenter gets the necessary controls to navigate between slides and stop the presentation.

Audience

When first opening the app, the users see a list of ongoing presentations. By selecting a presentation, they become part of the audience. The app shows the current slide of the presentation, and the screen is updated accordingly to the navigation of the presenter.

A user can at any moment leave the current session.

Technology

- Push messages are to be used for the control of the apps.
- Database/server: you may choose whatever technology for the server infrastructure.

REQUIREMENTS

1. A runnable prototype of the application implemented in Flutter, demonstrated on at least two real physical devices of your choice. [63pt + 5]
    a. Presenter functionality
        i. Initiating/stopping presentation (10pt)
        ii. Showing the slide (8pt)
        iii. Navigating through the presentation and updating the presenter's screen (15pt)
    b. Audience functionality
        i. Showing list of presentations (5pt)
        ii. Joining a presentation (10pt)
        iii. Screen update according to a.iii (15pt)
    c. Non-functional requirements [5pt]
        i. Define the NFRs for the project.
2. User testing and evaluation [8pt]
3. Project documentation [8pt]
    a. Description of the app explaining the different screens
    b. API documentation (URL, http method, parameters…)
    c. Results of user tests and findings (suggestions for improvements)
4. Handing over all required files in archive to lecturer before the beginning of the presentation session [4pt]

DEADLINES AND DELIVERABLES

Deadlines

1. April 24, 2023:
    a. User Interface design document containing wireframe and design template. Hand in through Ilias.

   b. ‘Coaching’ per group in Aachen G101 (no course session that day): face2face session about the project/questions. Schedule/timing will follow. Not mandatory but recommended.

2. June 5, 2023: ‘coaching’ per group in Aachen G101. Not mandatory but recommended

3. June 26, 2023: Project presentations. Everyone present!

Deliverables

- Digital project document describing the mobile app (functionality, user interface, configuration options…)
- Zip-archive containing the application program code and assets, video, documentation of API used.