

Abschlusspräsentation

Krawumm - Experimente für Kinder

Marlene Bauch, Jessica Eckardtsberg,
Johannes Otto, Michel Rost

“JOEMIA Group”

github.com/fh-erfurt/krawumm

Gliederung

- Projektidee
- Projektüberblick
- Klassendiagramm
- Kurzer Einblick in den Code
- Lessons learned
- Kommende Schritte

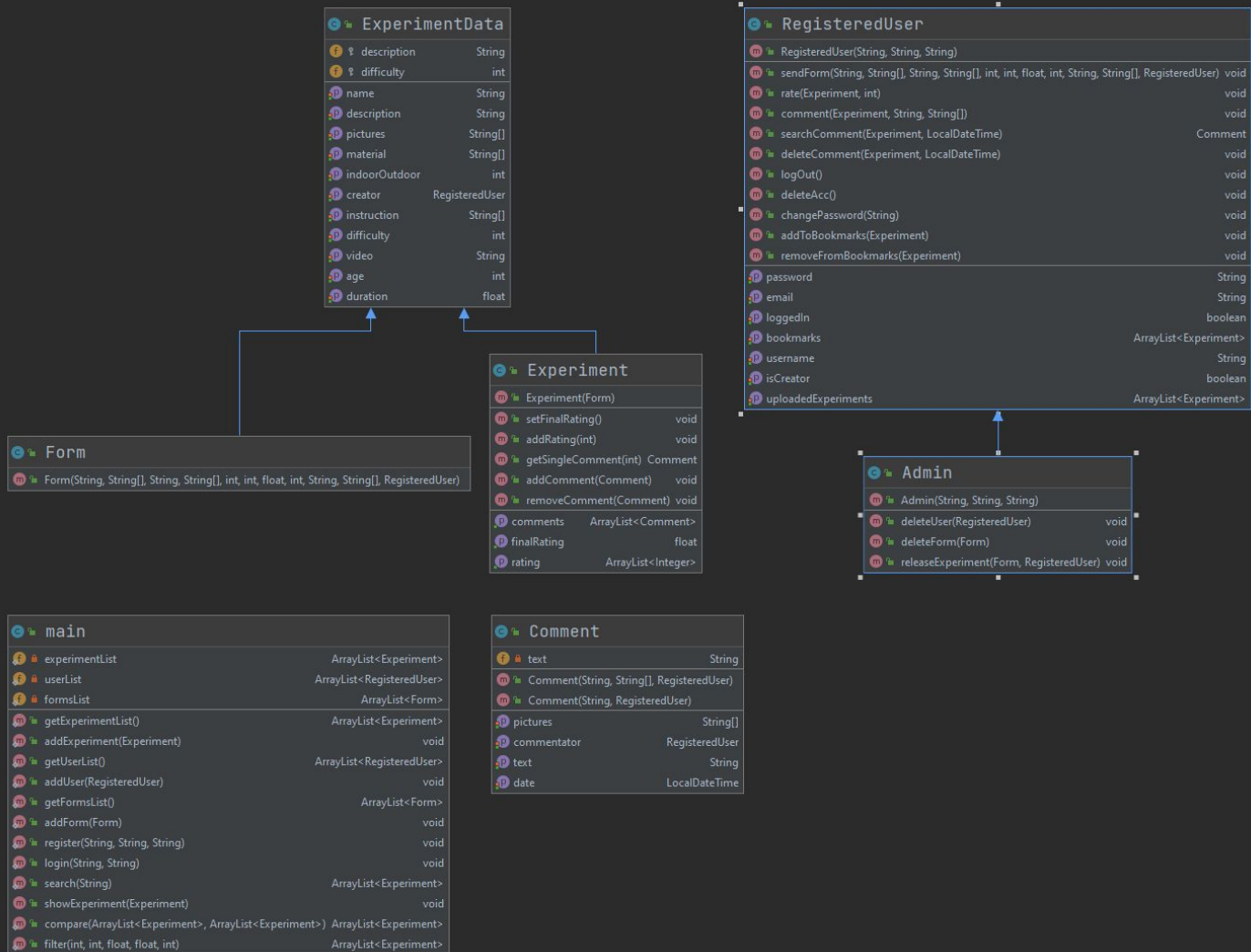
Projektidee

- Experimente für Kinder
- Interesse an Naturwissenschaften wecken
- Beschäftigung abseits von digitalen Medien
- Experimente mit (wenigen) Materialien, die man zu Hause hat
- Ideal für den Winter / eine Pandemie...

Projektüberblick

- Man kann Experimente nach Schlagworten suchen und nach bestimmten Kriterien filtern (z.B. Schwierigkeitsgrad)
- Als registrierter User kann man Experimente auch kommentieren und bewerten und eigene Experimente erstellen
- Experimente müssen von einem Admin bestätigt und veröffentlicht werden
- User können Einstellungen am Account vornehmen, eigene Kommentare löschen und Experimente zu ihrer Favoritenliste hinzufügen

Klassen- diagramm



Kurzer Einblick in den Code

```
/**
 * This method lets a user rate an experiment
 *
 * @param experiment is experiment the user rated
 * @param rating is the rating the user gave the experiment
 *
 * @see de.joemiagroup.krawumm.Experiment
 */
public void rate(Experiment experiment, int rating) {
    experiment.addRating(rating);
    experiment.setFinalRating();
}

/**
 * This method lets the user write a comment to an experiment
 * If pictures array is empty it uses the constructor without pictures and creates a comment without pictures
 *
 * @param experiment is the experiment the user commented
 * @param text is the content of the comment
 * @param pictures is an array of pictures for the comment
 */
public void comment(Experiment experiment, String text, String[] pictures) {
    if(pictures.length == 0){
        Comment comment = new Comment(text, registeredUser: this);
        experiment.addComment(comment);
    } else{
        Comment comment = new Comment(text, pictures, registeredUser: this);
        experiment.addComment(comment);
    }
}
```

Kurzer Einblick in den Code

```
@Test
void check_if_method_register_creates_a_new_user() {
    main.register( username: "Angler", email: "angler@email.de", password: "12345678");
    assertEquals( expected: "Angler", main.getUserList().get(0).getUsername(), message: "The user on this position should be the user with the name: Angler.");
}

@Test
void check_if_method_register_doesnt_create_a_new_account_if_email_already_exists() {
    main.register( username: "Jäger", email: "jaeger@email.de", password: "12345678");
    main.register( username: "Fischer", email: "jaeger@email.de", password: "12345678");
    assertEquals( unexpected: 3, main.getUserList().size(), message: "The user should not be added because the email already exists.");
}

@Test
void check_if_method_login_logs_into_the_right_account() {
    main.register( username: "Angler", email: "angler@email.de", password: "12345678");
    main.login( email: "angler@email.de", password: "12345678");
    assertEquals( expected: true, main.getUserList().get(0).isLoggedIn(), message: "The user on this position should be logged in!");
}
```

Lessons learned

- Bei Problemen können wir uns gegenseitig helfen, weil acht Augen mehr sehen als zwei
- Arraylisten können die Arbeit vereinfachen, aber auch deutlich erschweren
- Man kann Instanzen nicht so einfach löschen
- Pointer wie in C wären manchmal ganz nett
- Funktionen müssen static sein, um ohne Instanz darauf zugreifen zu können
- Gemeinsam programmieren macht mehr Spaß als alleine (wenn man nicht gerade tippen muss)
- Testen dauert mindestens genau so lange wie der Code an sich
- Elemente aus einer Liste zu löschen während man diese durchläuft, ist keine gute Idee

Kommende Schritte

- Auf Programmierung Java 2 warten und dann weiter machen.

Vielen Dank für die Aufmerksamkeit