Sprint 6 esitelmä

LiMaVi (Ryhmä 3)

- Suphitcha Pitkänen, Maria Aalto and Ville Schulz.

Aiheet

Sprint 6 sisältö

Projektin demo

Sprint 6 tavoitteet

Mitä seuraavaksi?

Haasteita?

Kysymyksiä ja Palautteita?

Sprint 6

Aloituspäivämäärä: 01.04.2025

Päättymispäivämäärä: 15.04.2025

Kesto: 2 viikkoa

• Työtunnit: Noin 15h/viikossa

Suunnitellut User Storyt: 9

Thteensä User Storyja: 9

Valmiit User Storyt: 9



Tiimiroolit

- ►Sprint 6:
- Scrum Master: Suphitcha Pitkänen
- ► Kehittäjä(backend): Maria Aalto
- ► Kehittäjä (frontend): Ville Schulz

Sprint 7:

Scrum Master: Maria Aalto

Kehittäjä(backend): Suphitcha Pitkänen

Kehittäjä(frontend): Ville Schulz

Statistical Code Review

```
public void drawNotes() {
   if (noteGrid == null) {
        noteGrid = new GridPane(10, 10);
       noteGrid.setPadding(new javafx.geometry.Insets(10, 0, 10, 0));
       noteArea.setContent(noteGrid);
    } else {
        noteGrid.getChildren().clear(); // Tyhjennä vanha sisältö
   Button addNoteButton = createAddButton();
   noteGrid.add(addNoteButton, 0, 0);
    int i = 1, j = 0;
    for (Note note : user.getNotes()) {
       Button noteButton = createNoteButton(note);
       noteButton.addEventHandler(MouseEvent.MOUSE_CLICKED, event -> openNoteView(note));
       noteGrid.add(noteButton, i, j);
       i++;
       if (i > 1) {
                                       //this method is used to draw the notes that match the search guery
           i = 0;
                                       private void drawFilteredNotes(List<Note> filteredNotes) {
            i++:
                                           noteGrid.getChildren().clear():
                                           if (filteredNotes.isEmpty()) {
                                              ImageView notFound = new ImageView(new Image("./images/NotFound.png"));
                                              notFound.setFitHeight(400);
                                              notFound.setFitWidth(400);
                                              notFound.getStyleClass().add("notFound");
                                              noteGrid.getChildren().clear();
                                              noteGrid.getChildren().add(notFound);
                                           int i = 0, j = 0;
                                          for (Note note : filteredNotes) {
                                              Button noteButton = createNoteButton(note);
                                              noteButton.addEventHandler(MouseEvent.MOUSE_CLICKED, event -> openNoteView(note));
                                              noteGrid.add(noteButton, i, j);
                                              i++;
```

```
117 >
            public void drawNotes() {
                if (noteGrid == null) {
                    noteGrid = new GridPane(10, 10);
                    noteGrid.setPadding(new javafx.geometry.Insets(10, 0, 10, 0));
                    noteArea.setContent(noteGrid):
               } else {
                    noteGrid.getChildren().clear(); // Tyhjennä vanha sisältö
                Button addNoteButton = createAddButton():
                noteGrid.add(addNoteButton, 0, 0);
                int i = 1;
                int j = 0;
                for (NoteNode note : noteNodes) {
                    noteGrid.add(note.getNoteButton(), i, j);
                    i++:
                    if (i > 1) {
                        i = 0;
                        j++;
            //this method is used to draw the notes that match the search query
142 ∨
            private void drawFilteredNotes(List<NoteNode> filteredNotes) {
                noteGrid.getChildren().clear():
                if (filteredNotes.isEmpty()) {
                    ImageView notFound = new ImageView(new Image("./images/NotFound.png"));
                    notFound.setFitHeight(400);
                    notFound.setFitWidth(400);
                    notFound.getStyleClass().add("notFound");
                    noteGrid.getChildren().clear();
                    noteGrid.getChildren().add(notFound);
                    return;
```

Ratkaisu ja optimointi

Alkuperäinen note noden luonti oli erittäin huonosti optimoitu. Joka kerta kun käyttäjö kirjoitti search-kenttään hakuehdon niin ohjelma etsi kyseisellä nimellä noteja ja loi sen pohjalta uudet graaffiset elementit. Nytten uudessa versiossa joka notesta luodaan valmiiksi graaffiset elementit muistiin, joita sitten haetaan kun suoritetaan searchfunktio

Checkstyle and SpotBugs

```
+ import datasource.ImageHandling;
+ import datasource.NoteDAO;
+ import datasource.UserDAO;
import model.Note;
import model.User;
- import datasource.UserDAO;
- import datasource.NoteDAO;
- import datasource.ImageHandling;
```

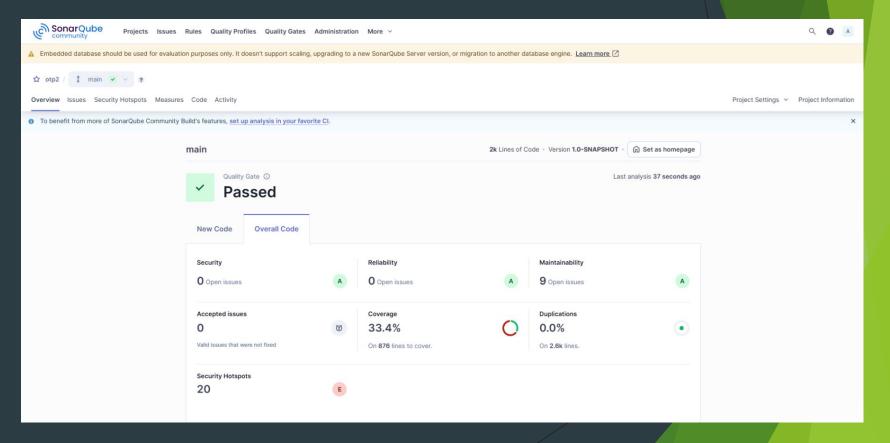
```
- notes=user.getNotes();
+ notes=user.getNotes();
- Image profilePicture = new Image(profilePictureUrl);
- return profilePicture;
+ return new Image(profilePictureUrl);
```

```
(400' is a magic number. (146:35) [MagicNumber] <Sun Checks>
(400' is a magic number. (147:34) [MagicNumber] <Sun Checks>
```

- Olace MainCareanController lacks like decianed for extension foon he

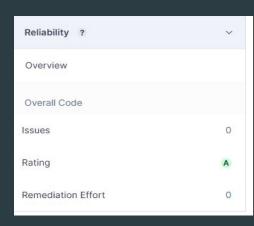
```
<plugin>
    <artifactId>maven-javadoc-plugin</artifactId>
    <version>3.6.3</version>
    <configuration>
        <source>21</source>
        </configuration>
        </plugin>
```

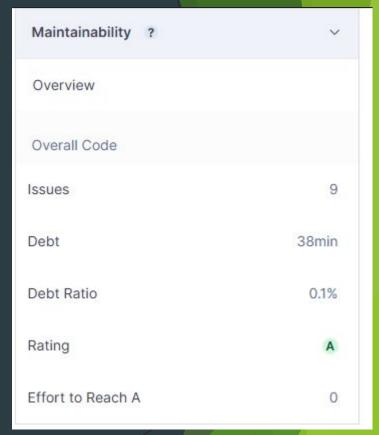
SonarQube



Size	~
Lines of Code	1,992
Lines	2,601
Statements	731
Functions	154
Classes	23
Files	25
Comment Lines	160
Comments (%)	7.4%
Complexity ?	~
Cyclomatic Complexity	238
Cognitive Complexity	142

~
0
Α
0





Acceptance testing plan

Functional Criteria:

- Users can register and log in securely.
- Users can update their profile information.
- Users can create, edit, and delete notes.
- Users can search for notes using keywords.
- Notes are saved persistently in the MariaDB database.
- The application follows the MVC structure and behaves consistently with user actions.

Usability Criteria:

- The UI is clean, responsive, and easy to navigate.
- All buttons and actions are clearly labeled and logically placed
- Users receive appropriate feedback (e.g., success messages, validation errors).



Projektin

Demo

Sprint 6 tavoitteet





Dokumentointi



Tuotteen piirteet:

- Koodi vie vähemmän muistia

Seuraavaksi

- Toiminnallinen testaus ja Ei-toiminnallinen testausta
- Sprint 7 suunnitteleminen
- Projektissa:
 - Testaa, että järjestelmä toimii oikein eri syöteillä
 - Tietoturvatestaus



Haasteita?

►Kaunis ilma ulkona. Haasteita olla sisällä koodaamassa.



Kysymyksiä ja Palautteita?