STUDYFLOW

STUDY PLANNER

Hilda Hermunen,
Veera Ruotsalainen

Group 7

Introduction

- Study planner for busy students like ourselves.
- AI tools were used for debugging, suggestions and refactoring.
  - Used AI tools: GitHub Copilot, ChatGPT
- GitHub link: Linkki
- Trello link: Linkki

Product Backlog

- Create database
- Create models for the project
- Create controllers for the project
- Create database connection
- Create UI/UX design
- Apply functionality
- Create unit tests for the project
- Apply testing automation
- Coverage testing
- Create project documentation
- Final deployment

Sprint 1

- Sprint Master: Veera
- Sprint Backlog: Project vision, plan and scheduling, low-level Figma prototype, database and environment setup
- Done: Project vision, plan and scheduling, low-level Figma prototype
- Release: Project vision, plan, low-level Figma prototype
- Postpone: Finishing database and environment setup

Sprint 2

- Sprint Master: Hilda
- Sprint Backlog: Creating database , necessary models and controllers, daos and database connection. Applying unit testing, designing UI/UX. Apply basic functionality of the product.
- Done: The database, models, controllers and the database connection was created. Unit testing was applied and UI/UX was designed. Most of the basic functionality was applied.
- Release: A working prototype with database connectivity, UI design and core features. Users can create courses, study sessions, assignments, and exams.
- Postpone: Functionality where user can integrate their class schedule with the study planner and delete tasks from the planner.
- Testing: Unit tests were run on models.

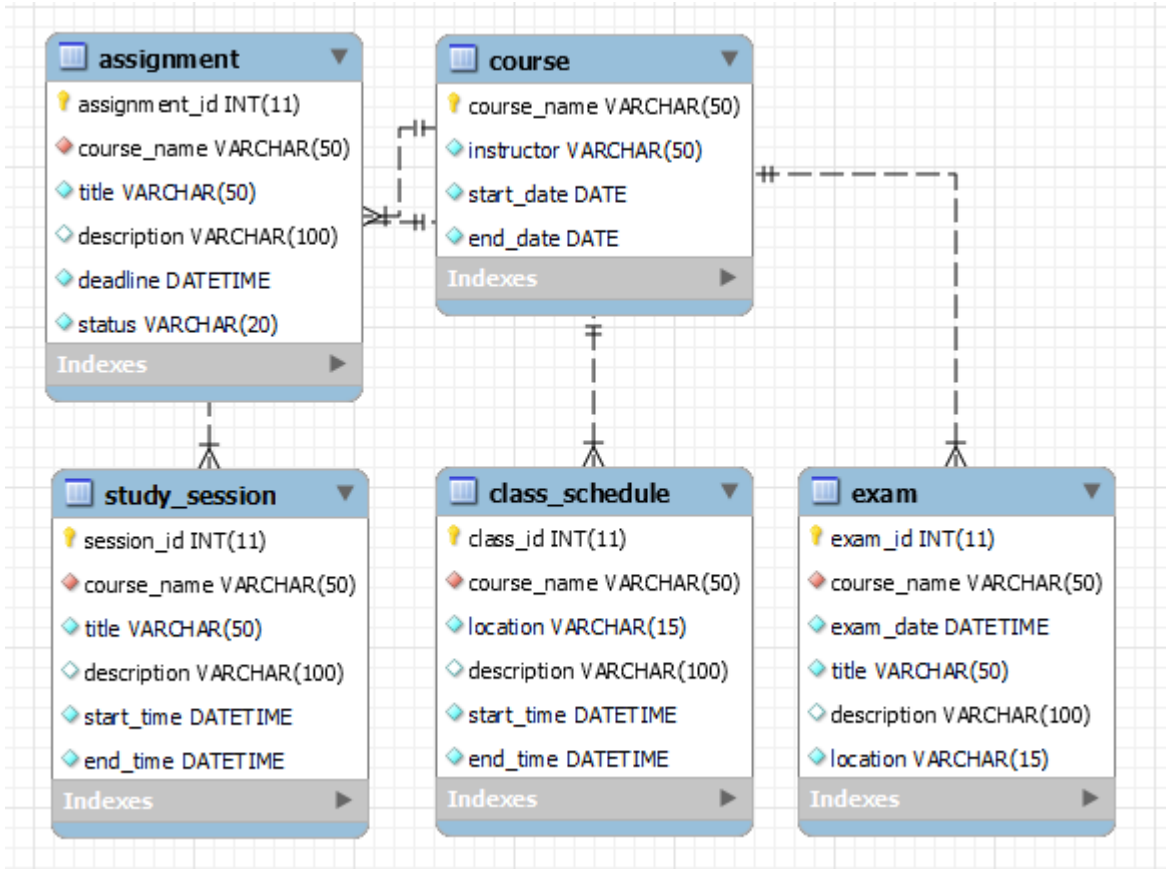
Sprint 3

- Sprint Master: Veera
- Sprint Backlog: Finishing the functionalities of the project, including updating and deleting assignments in the timetable and integrating class schedule. CSS and FXML fixes. Docker image and Jenkins.
- Done: All above
- Release: Stable version of the application with full functionality and refined UI/UX.
- Postpone: No backlogs postponed in this sprint.
- Testing: Automated tests were implemented using Jenkins.

Sprint 4

- Sprint Master: Hilda
- Sprint Backlog: Enhance CSS, Cleaning code and refactoring, Project presentation, Final deployment
- Done: All above
- Release: The final version of the application was deployed, and the project is ready to be presented with key functionalities.
- Testing: Coverage testing was applied using JaCoCo.

Architectural Design



Database EER Diagram made with MySQL Workbench

Applied Technologies

UI/UX prototype – Figma

UI/UX design – Scene Builder

Database – MariaDB (SQL)

Backend – Java

Frontend – JavaFX, FXML

Styling – CSS

Database connection – JDBC

Dependency management – Maven

Testing automation – Jenkins

Code coverage testing – JaCoCo

Deployment – Docker

Demo

- 5 minutes maximum
  - Prototype demo

Learning achievement

- Using new technologies JaCoCo, Jenkins and Docker.
- Good communication and careful planning are key to a successful project.
  - Even when challenges came up, teamwork and good planning helped us stay focused and complete project successfully.
- Deepened our understanding of Scrum and its benefits in project management.

Plan for further development

- Continuing the project on SEP2
  - Reminders, user login, filtering events?
- Mockito
- Docker deployment with JavaFX