►ZHAW MAS I 13 Programmieren Fallbeispiel VereinsManagerPro

Bernhard Kämpf & Serge Kaulitz

ZHAW MAS I 13 Programmieren Fallbeispiel VereinsManagerPro

Bernhard Kämpf & Serge Kaulitz

Das Java-Projekt hat den Fokus, verschiedene Paradigmen und Technologien in der Praxis umzusetzen. Das Vorgehen wurde nach dem Best Practice Ansatz angegangen. Folgende der im Modul behandelten Aspekte bzw. Sprachelemente wurden eingesetzt:

- JavaFX (mit Scene Builder)
- File I/O (inkl. Serialisierung)
- Threads (Observer-Entwurfsmuster)

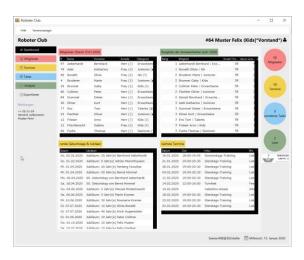


Bild: Dashboard VereinsManagerPro©

Planung

Zuerst wurden Ideen über Möglichkeiten und Umfang diskutiert. Dazu hat das Projektteam eine Roadmap und Scope definiert. Die Applikation sollte Modular aufgebaut werden und einen Nutzen haben, den beide Teilnehmer der Gruppe gebrauchen können.

Umsetzung

In der Umsetzung wurde vom Konzept bis zur fertigen Applikation in Phasen und iterativ gearbeitet. Das Design orientierte sich an Mockups, in denen die Anordnung der einzelnen Funktionen und die Gestaltung der Ansichten entworfen wurden.

Applikations-Module

Login

Das Login gewährleistet den Zugriff für Vereinsmitglieder. Dazu wurde ein Framework mit Login und Registrierung erstellt.

Dashboard

Das Dashboard dient zur allgemeinen Übersicht über die Mitglieder und anstehenden Projekte respektive Termine. Die Menü- und Navigationselemente wurden in einem Rahmen um das Dashboard herum platziert. Auf der rechten Seite befinden sich Key Figures.

Mitglieder

Das Modul «Mitglieder» stellt die Funktionalitäten zur Bewirtschaftung der Vereinsmitglieder bereit. Die Mitgliederdaten sind unterteilt in Grunddaten, Kontaktdaten und Infos. Für die Mitgliederliste stehen Exportfunktionen zur Verfügung (Ausgabe als PDF- oder Excel-Files). Im Suchfeld kann nach Vorname und Name gefiltert werden.

Termine

Im Termin-Modul können neue Termine erfasst und mit allen notwendigen Detailangaben ergänzt werden. Mitglieder können sich über ein Web-Formular für Termine an- oder abmelden. Die Präsenzkontrolle erfolgt ebenfalls über ein Web-Formular.

Tasks

In den Tasks kann der Benutzer die anstehenden Aufgaben bewirtschaften und überwachen. Die Aufgaben werden in der Anzeige entsprechend ihrem Status gruppiert (TreeView): Überfällig, Pendent, In Arbeit, Erledigt.

Analyse

In diesem Menu gibt es verschiedene Möglichkeiten, Daten auszuwerten oder als Diagramme darzustellen: Mitglieder nach Kategorie, Anzahl Geburtstage pro Monat und Tasks nach Status.

Datenbank

Die Applikation VereinsManagerPro© verfügt über eine Datenbank-Anbindung (MySOL). Die Bewirtschaftung der Tabelleninhalte (z.B. Mitgliederdaten) erfolgt innerhalb der Applikation über JavaFX-User Interfaces (mit Login). Daneben wurden webbasierte Formulare (PHP und HTML) implementiert, über die sich

Mitglieder ohne Zugang zur Applikation über den Browser ihres Smartphones/Tablets für Termine anoder abmelden können (vergleichbar mit Doodle). Um die Daten zwischen Datenbank und der Applikation synchron zu halten wurde ein Observer-Muster implementiert.

Review

In der Umsetzung gab es viele Herausforderungen. Hier eine nicht abschliessende Liste von Beispielen.

Debugging

Das Debugging hat dabei geholfen, Probleme im Bereich der Performance einzugrenzen und die Applikation diesbezüglich zu optimieren. Auch die anfänglich häufig aufgetretenen Nullpointer-Exceptions konnten auf diese Weise eliminiert werden.

Git

Die Zusammenarbeit mit Git soll die Entwicklungsarbeit unterstützen und vereinfachen. In der Praxis liess sich dies nicht so einfach bewerkstelligen. Aufgrund der fehlenden Erfahrung gab es immer wieder unerwartetes Verhalten mit Commits von Branches. Mehrmals musste das Repository komplett neu implementiert werden.

Supportfunktionen

In der Top-Navigation wurden Supportfunktionen und Informationen zur Verfügung gestellt: Online-Hilfe, Linksammlung, JavaDoc, Anzeige LogFile, Info, Benutzerliste usw.

Unterschiedliche Betriebssysteme

Intellij als IDE konnte problemlos auf OS Mac und Windows eingesetzt werden. Jedoch verhält sich die Applikation zur Laufzeit auf Mac teilweise anders als auf Windows (z.B. Toleranz bei nicht geschlossenem InputStream [close()] oder in der Verarbeitung und Priorisierung von Threads). Der dazugehörige Code musste für Windows angepasst werden.

Testing

Die Applikation wurde einerseits mit JUnit-Tests und anderseits mit User-Test und Abnahmekriterien getestet. Viele Fehler konnten so, neben dem normalen Debuggen, behoben werden.

Element	Class, %	Method, %
controller ===================================	50% (4/8)	36% (32/88)
einstellungen	100% (2/2)	100% (15/15)
model model	95% (19/20)	56% (123/217)
services	42% (3/7)	20% (16/78)
view view	0% (0/48)	0% (0/269)

Tabelle: Übersicht JUnit-Test-Coverage

To Do

Die für das MVP vorgesehenen Funktionen konnten vollumfänglich implementiert werden, wie auch einige der angestrebten Erweiterungen. Optimierungen innerhalb der Funktionen des MVP und die Realisierung weiterer Module könnten in einer späteren Version erfolgen. Ein paar Beispiele für Bereiche mit Verbesserungspotential:

- Performanceverbesserungen und Beseitigung bekannter Bugs
- Deployment (ausführbares JAR-File)
- Wording und Design
- UX
- Logikfehler

Artefakte

- <u>lavaDoc</u>
- <u>Java-Code auf GitHub</u>
- PHP und HTML-Element auf GitHub
- Online-Hilfe
- Datenblatt für DB-Anbindung
- Mockups

Facts

Ein paar Facts zum Projekt VereinsManagerPro©:

- 7'500 Zeilen Code
- 5 Hauptfunktionen
- Design und Entwicklung iterativ
- Java und JavaFX
- Intelij auf Mac und Win