DHBW
Duale Hochschule
Baden-Württemberg
Stuttgart

# Project Manual

(TINF21C, SWE)

**Project:** Modelling Wizard Improvements

**Customer:** Markus Rentschler

Christian Holder


**Team:**

| | |
|---|---|
| Project Manager | – Robin Ziegler (inf21100@lehre.dhbw-stuttgart.de) |
| Developer | – Nils Hoffmann (inf21194@lehre.dhbw-stuttgart.de) |
| Test Manager | – Michael Grote (inf21111@lehre.dhbw-stuttgart.de) |
| System Architect | – Fabian Kreuzer (inf21106@lehre.dhbw-stuttgart.de) |
| Tech. Documentation | – Dana Frey (inf21099@lehre.dhbw-stuttgart.de) |
| Product Manager | – Maximilian Trumpp (inf21123@lehre.dhbw-stuttgart.de) |

**Change History**

| Version | Date | Author | Comment |
|---|---|---|---|
| 0.1 | 22.09.2022 | Michael Grote | Preliminary Setup |
| 1.0 | 06.10.2022 | Michael Grote | Edit Content |

# Table of Contents

# 1. Project Assignment

<table>
<tr><th colspan="2">Project Assignment</th></tr>
<tr><td colspan="2">Project Objective (Output)<br>The aim of the project is to analyze the usability of the existing Windows stand-alone application “Modelling Wizard”. From this, a usability concept for the GUI is to be developed. In addition, the existing functions are to be tested. Furthermore, the existing source code is to be refactored.</td></tr>
<tr><td colspan="2">Project Benefit (Outcome)<br>The user-friendliness is to be improved by adjustments to the GUI. Thus, the user should later be able to use the software more intuitively. In addition, already known errors should be eliminated. By refactoring the existing source code, maintainability should be improved. This also results in the preservation of adaptability for the future.</td></tr>
<tr><td>Customer:<br>M. Rentschler; C. Holder</td><td>Project leader:<br>Robin Ziegler</td></tr>
<tr><td>Team members:<br>• Robin Ziegler<br>• Maximilian Trumpp<br>• Michael Grote<br>• Fabian Kreuzer<br>• Dana Frey<br>• Nils Hoffmann</td><td>Previous developer team:<br>• Linus Eickhoff<br>• Florian Kellermann<br>• Lukas Ernst<br>• Florian Kaiser<br>• Malte Horst<br>• Rajkumar Pulaparthi</td></tr>
<tr><td>Main tasks:<br>• Documentation<br>• Analysis<br>• Design<br>• Development<br>• Create tests</td><td>Milestones:<br>• Analysis phase<br>• Design phase<br>• Coding<br>• Test phase<br>• Presentation</td></tr>
<tr><td>Project start event:<br>Introductory lecture with project assignment</td><td>Project start date:<br>09.September 2022</td></tr>
<tr><td>Project end event:<br>Presentation of the Results</td><td>Project end date:<br>14.May 2023</td></tr>
</table>

## 2. Project Context

| Initial situation and problem description |
|---|
| The current implementation has minor bugs and still needs a more detailed bug analysis. In addition, the user interface has several buttons with similar functions. This reduces the user friendliness. Also, some menus should be changed from the “Easy Mode” to the “Advanced Mode” to simplify the usage in the Easy Mode. |

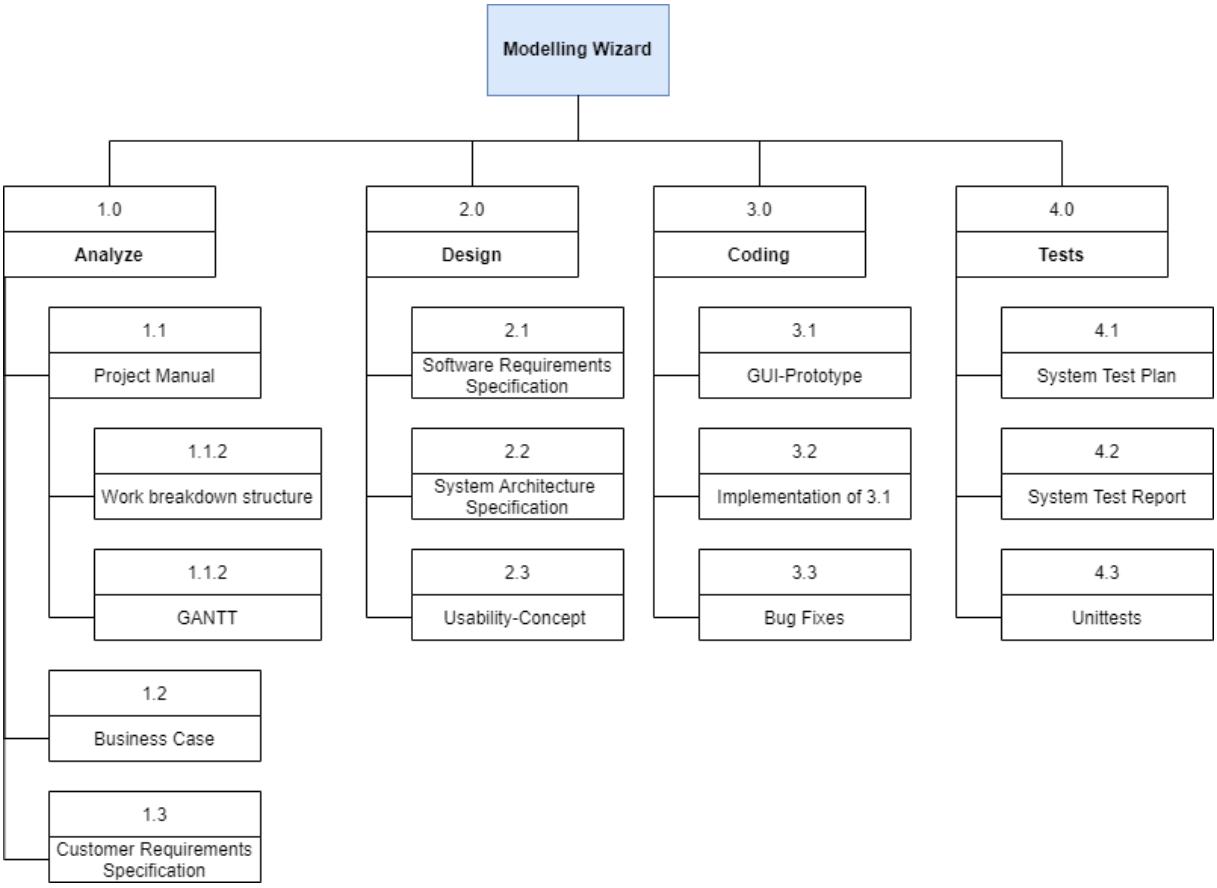
| Temporal Project Context | |
|---|---|
| Pre-project phase | Post-project phase |
| • The stand-alone application has already been developed by another team. The GUI has some small bugs, and the user-friendliness can be increased even further | • The existing GUI should be redesigned. The existing code should be refactored. New tests should be implemented for the software. |

| Stakeholder Analysis | | | |
|---|---|---|---|
| Stakeholder | Potential / Chance | Conflict / Risks | Actions |
| Customer | Satisfaction with the new implementation. | Requests for changes during the project. | Regular communication between supplier and client. |
| Supplier | Development of a solution that meets the requirements. | Misjudgment of effort, time pressure, miscommunication | Fixed intermediate dates, regular meetings, uniform code standards |
| User | Benefit of the application, increase in efficiency, intuitive usability | Lack of understanding of the application, incorrect operations | Creation of a usability concept and testing of catching errors |

## 3. Project Organization

| Project Organization | | |
|---|---|---|
| Position | Description | Name |
| Customer | • Defines the project framework<br>• Sets out the customer requirements | Rentschler, Markus / Holder, Christian |
| Project leader | • Control of the distribution of tasks<br>• Coordination of the members<br>• Allocation of resources | Robin Ziegler |
| Project team | Product Manger<br>Test Manager<br>System Architect<br>Tech. Documentation<br>Developer | Maximilian Trumpp<br>Michael Grote<br>Fabian Kreuzer<br>Dana Frey<br>Nils Hoffmann |

# 4. Work Breakdown Structure (PSP)



## 5. Milestones

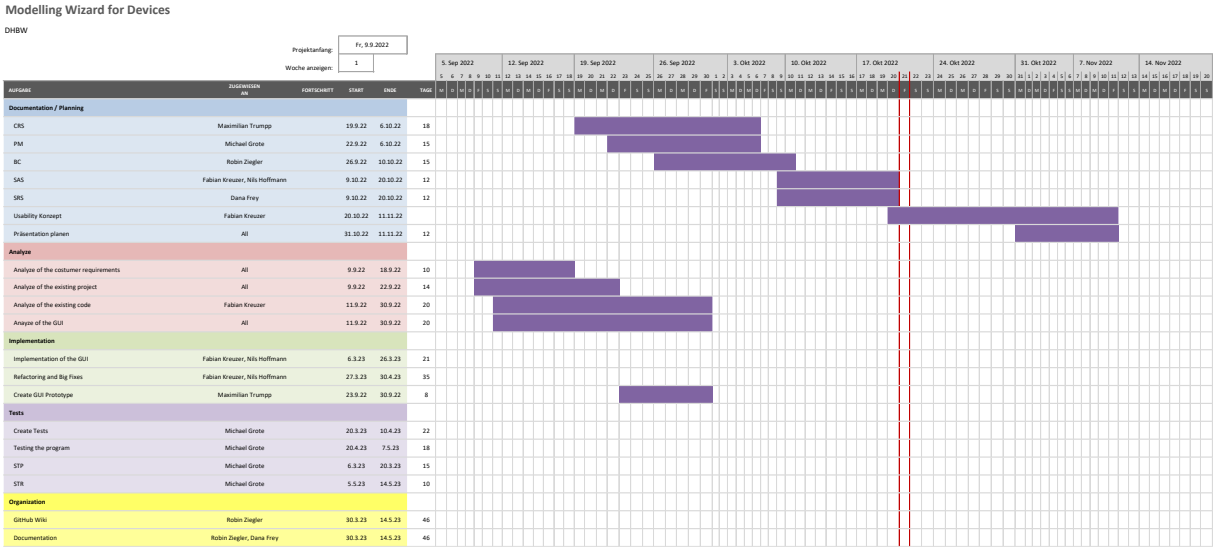
| WP-Code | Milestone Name | Plan Date | Responsible Person |
|---|---|---|---|
| 1.0 | Analyze | | |
| 1.1 | Project Manual | 06.10.2022 | Michael Grote |
| 1.1.1 | Work breakdown structure | 06.10.2022 | Michael Grote |
| 1.1.2 | GANTT | 06.10.2022 | Michael Grote |
| 1.2 | Business Case | 10.10.2022 | Robin Ziegler |
| 1.3 | Customer Requirements Specification | 06.10.2022 | Maximilian Trumpp |
| 2.0 | Design | | |
| 2.1 | Software Requirements Specification | 20.10.2022 | Dana Frey |
| 2.2 | System Architecture Specification | 20.10.2022 | Fabian Kreuzer, Nils Hoffmann |
| 2.3 | Usability-Concept | 11.11.2022 | Fabian Kreuzer |
| 3.0 | Coding | | |
| 3.1 | GUI-Prototype | 30.09.2023 | Maximilian Trumpp |
| 3.2 | Implementation of 3.1 | 30.03.2023 | Fabian Kreuzer |
| 3.3 | Bug Fixes | 30.04.2023 | Fabian Kreuzer |
| 4.0 | Tests | | |
| 4.1 | System Test Plan | 20.03.2023 | Michael Grote |
| 4.2 | System Test Report | 14.05.2023 | Michael Grote |
| 4.3 | Unittests | 10.04.2023 | Michael Grote |

## 6. Lists of Tasks and responsible person

| Activities and Responsibility | | |
|---|---|---|
| Person | Category | Tasks |
| **Robin Ziegler**<br>*Position:* Project Leader<br>*GitHub-Name:*<br>robinziegler | Documentation, Organization | • Business Case<br>• Organization of GitHub<br>• Contact person to the Customer |
| **Maximilian Trumpp**<br>*Position:* Product Manager<br>*GitHub-Name:*<br>maximiliantrumpp | Documentation | • Customer Requirements Specification<br>• Contact person to the Customer |
| **Michael Grote**<br>*Position:* Test Manager<br>*GitHub-Name:* michi3214 | Documentation | • Project Manual |
| | Development | • Unittests |
| **Fabian Kreuzer**<br>*Position:* System Architect<br>*GitHub-Name:*<br>Fabiankreuzer | Documentation | • System Architecture Specification |
| | Development | • Implementation of the new GUI |
| **Dana Frey**<br>*Position:* Tech. Documentation<br>*GitHub-Name:* DanaFrey | Documentation | • Software Requirements Specification<br>• Meeting Minutes |
| **Nils Hoffmann**<br>*Position:* Developer<br>*GitHub-Name:*<br>HoffmannNils | Documentation | • System Architecture Specification |
| | Development | • Implementation of the new GUI |

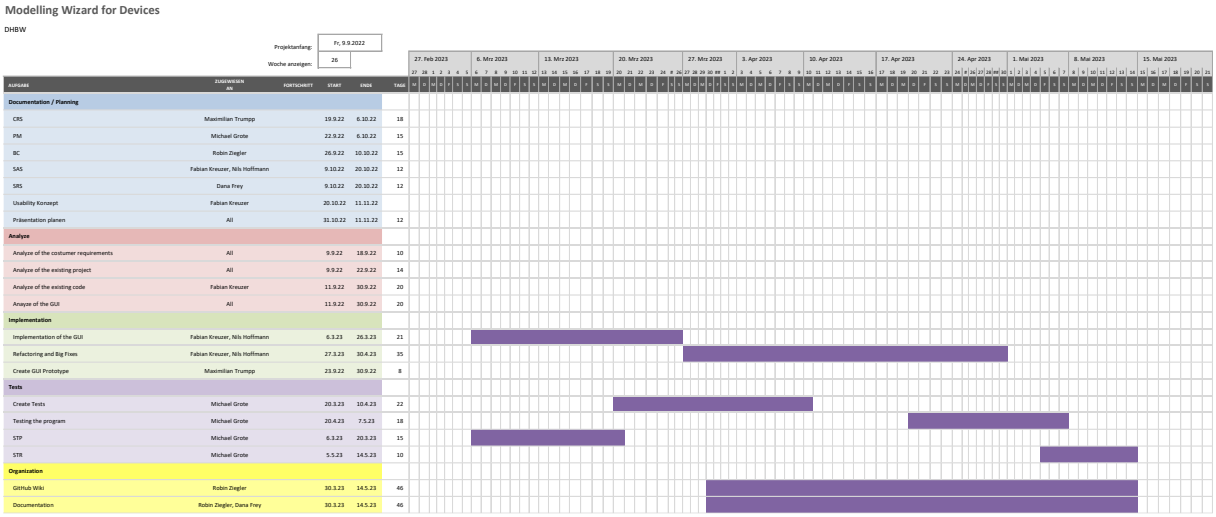
# 7. Gant-Chart

First part of the project:




| AUFGABE | ZUGEWIESEN AN | FORTSCHRITT | START | ENDE | TAGE |
|---|---|---|---|---|---|
| **Documentation / Planning** | | | | | |
| CRS | Maximilian Trumpp | | 19.9.22 | 6.10.22 | 18 |
| PM | Michael Grote | | 22.9.22 | 6.10.22 | 15 |
| BC | Robin Ziegler | | 26.9.22 | 10.10.22 | 15 |
| SAS | Fabian Kreuzer, Nils Hoffmann | | 9.10.22 | 20.10.22 | 12 |
| SRS | Dana Frey | | 9.10.22 | 20.10.22 | 12 |
| Usability Konzept | Fabian Kreuzer | | 20.10.22 | 11.11.22 | |
| Präsentation planen | All | | 31.10.22 | 11.11.22 | 12 |
| **Analyze** | | | | | |
| Analyze of the costumer requirements | All | | 9.9.22 | 18.9.22 | 10 |
| Analyze of the existing project | All | | 9.9.22 | 22.9.22 | 14 |
| Analyze of the existing code | Fabian Kreuzer | | 11.9.22 | 30.9.22 | 20 |
| Anayze of the GUI | All | | 11.9.22 | 30.9.22 | 20 |
| **Implementation** | | | | | |
| Implementation of the GUI | Fabian Kreuzer, Nils Hoffmann | | 6.3.23 | 26.3.23 | 21 |
| Refactoring and Big Fixes | Fabian Kreuzer, Nils Hoffmann | | 27.3.23 | 30.4.23 | 35 |
| Create GUI Prototype | Maximilian Trumpp | | 23.9.22 | 30.9.22 | 8 |
| **Tests** | | | | | |
| Create Tests | Michael Grote | | 20.3.23 | 10.4.23 | 22 |
| Testing the program | Michael Grote | | 20.4.23 | 7.5.23 | 18 |
| STP | Michael Grote | | 6.3.23 | 20.3.23 | 15 |
| STR | Michael Grote | | 5.5.23 | 14.5.23 | 10 |
| **Organization** | | | | | |
| GitHub Wiki | Robin Ziegler | | 30.3.23 | 14.5.23 | 46 |
| Documentation | Robin Ziegler, Dana Frey | | 30.3.23 | 14.5.23 | 46 |

Second part of the project:




| AUFGABE | ZUGEWIESEN AN | FORTSCHRITT | START | ENDE | TAGE |
|---|---|---|---|---|---|
| **Documentation / Planning** | | | | | |
| CRS | Maximilian Trumpp | | 19.9.22 | 6.10.22 | 18 |
| PM | Michael Grote | | 22.9.22 | 6.10.22 | 15 |
| BC | Robin Ziegler | | 26.9.22 | 10.10.22 | 15 |
| SAS | Fabian Kreuzer, Nils Hoffmann | | 9.10.22 | 20.10.22 | 12 |
| SRS | Dana Frey | | 9.10.22 | 20.10.22 | 12 |
| Usability Konzept | Fabian Kreuzer | | 20.10.22 | 11.11.22 | |
| Präsentation planen | All | | 31.10.22 | 11.11.22 | 12 |
| **Analyze** | | | | | |
| Analyze of the costumer requirements | All | | 9.9.22 | 18.9.22 | 10 |
| Analyze of the existing project | All | | 9.9.22 | 22.9.22 | 14 |
| Analyze of the existing code | Fabian Kreuzer | | 11.9.22 | 30.9.22 | 20 |
| Anayze of the GUI | All | | 11.9.22 | 30.9.22 | 20 |
| **Implementation** | | | | | |
| Implementation of the GUI | Fabian Kreuzer, Nils Hoffmann | | 6.3.23 | 26.3.23 | 21 |
| Refactoring and Big Fixes | Fabian Kreuzer, Nils Hoffmann | | 27.3.23 | 30.4.23 | 35 |
| Create GUI Prototype | Maximilian Trumpp | | 23.9.22 | 30.9.22 | 8 |
| **Tests** | | | | | |
| Create Tests | Michael Grote | | 20.3.23 | 10.4.23 | 22 |
| Testing the program | Michael Grote | | 20.4.23 | 7.5.23 | 18 |
| STP | Michael Grote | | 6.3.23 | 20.3.23 | 15 |
| STR | Michael Grote | | 5.5.23 | 14.5.23 | 10 |
| **Organization** | | | | | |
| GitHub Wiki | Robin Ziegler | | 30.3.23 | 14.5.23 | 46 |
| Documentation | Robin Ziegler, Dana Frey | | 30.3.23 | 14.5.23 | 46 |

## 8. Risks

- ***Financial Risk:***
  Due to delays in development or unforeseen bugs, the development effort could increase. The higher development effort would increase personal costs.

  ***Actions:***
  By distributing development tasks among different team members, some processes can be performed more efficiently.

- ***Planning Risk:***
  Time schedules might have been planned too tightly.

  ***Actions:***
  For unexpected development efforts, a team member should act as a floater. In addition, certain buffers should be built into the schedules.

- ***Technical Risk:***
  Since development has already been started by a previous team, our team needs to get familiar with the existing code first. The programming language C# is only known to a limited extent in our team.

  ***Actions:***
  A good familiarization with the existing project and the used programming language is necessary. In this context it is useful to get used to the existing documentation.

- ***Legal Risk:***
  Due to the use of software from other developers, attention to possible legal conditions is necessary.

  ***Actions:***
  When using components from other developers, attention must be paid to possible licenses and plagiarism must be avoided as much as possible.

- ***Personal Risk:***
  Significant problems could arise due to possible staff absences. The spontaneous loss of a team member would compromise the scheduled completion.

  ***Actions:***
  By distributing development among multiple team members, the impact of a team member leaving can be better hedged.