

# Abstract

Projektname: ProductivityGarden

Name

Jonas Huber

Matrikelnummer

IU14085128

Modul

Projekt: Software Engineering

DLMCSPSE01\_D

Datum

25.01.2025

# 1. Making-of

## **Projektidee**

Die Grundidee von ProductivityGarden basiert auf der bekannten Pomodoro-Technik, erweitert um ein spielerisches Belohnungssystem. Die Idee ist entstanden, weil ich selbst ein großer Fan von Produktivitätssteigerungs-Anwendungen und Projekt- bzw. Zeittracking bin. Aus diesem Grund habe ich in der Vergangenheit bereits mehrfach ähnliche Anwendungen genutzt.

## **Planung und Entwurf**

Zu Beginn wurden User-Stories und erste UML-Diagramme (z.B. Klassendiagramm) entwickelt. Aufgrund mangelnder Erfahrung gestaltete sich dieser Entwurfsprozess jedoch herausfordernd. Zudem erwies sich die Definition einer klaren Architektur als anspruchsvoll, was später zu mehrfachen Anpassungen im Code führte.

## **Aufteilung der Implementierung**

Die Implementierungsphase wurde in drei weitere Phasen aufgeteilt. Zunächst wurde an der Implementierung des Dashboards, der Zeiterfassung & des Punktekontos gearbeitet. Diese haben sich als die wichtigsten Komponenten herausgestellt und sollten daher als erstes implementiert und getestet werden. Danach wurden die Features rund um die Projekte implementiert und getestet. Nach Fertigstellung der produktiven Anwendung, konnte mit der Implementierung der Spielkomponente begonnen werden.

## **Spielkomponente und Assets**

Für die Gestaltung der Garten-Objekte wurde zunächst mit *Pillow* experimentiert. Schließlich wurden die Bilder mit *DALL·E* generiert und der Hintergrund mit *GIMP* entfernt, um die Pflanzen und Dekorationen optisch besser in die Gärten zu integrieren.

# 2. Kritische Reflexion

## Was lief gut?

### **Motivierende Projektidee:**

Da ich selbst zur Zielgruppe der Anwendung gehöre, war das Projekt insgesamt sehr praxisnah und interessant.

### **Implementierung aller grundlegenden Funktionalitäten:**

Die in der Konzeptionsphase (hauptsächlich im Anforderungs- und Spezifikationsdokument) definierten Features und Geschäftsregeln konnten fast ausnahmslos erfolgreich umgesetzt werden.

### **Technische Umsetzung:**

Die Verwendung von PyQt6 und Pygame bot eine solide Basis, um sowohl GUI als auch spielerische Elemente ohne Vorkenntnisse zu realisieren.

## Herausforderungen und Verbesserungsmöglichkeiten

### **Entwurf und Architektur**

Durch fehlende Vorerfahrungen gestaltete sich die Konzeptionsphase schwieriger als erwartet. In der Folge wurde bspw. der Code sowohl in der *session.py* als auch in der *virtualgarden.py* stark gebündelt. Dadurch besteht keine klare Trennung zwischen den verschiedenen Schichten in der

Schichtenarchitektur. Künftig sollte eine bessere Trennung der Schichten erfolgen. Beispielsweise könnte man sich stärker am MVC Design Pattern orientieren.

### **Zeitplan und Umsetzung**

Die Projektphasen waren zu wenig klar auf Basis- und Erweiterungsfunktionen abgestimmt. Es empfiehlt sich, Implementierungsphasen deutlicher in die Kategorien „Essentials“ und „Optimizations/Optionals“ zu trennen. Dies hätte es ermöglicht, die Kernfunktionen stabil und vollständig fertig zu stellen, bevor optionale Funktionen integriert oder Optimierungen z.B. am GUI-Design vorgenommen werden.

Da die Architekturphase zu kurz kam, mussten spätere Anpassungen vorgenommen werden. Das führte zu verringertem Fokus auf Tests. Eine engmaschigere Projektüberwachung und ein flexibleres Task-Management (z.B. Scrum-Sprints) hätten hier helfen können.

### **Testing**

Auch wenn das Projekt ein funktionsfähiges Produkt liefert, wurde das Testkonzept aus Zeitgründen vernachlässigt. Besonders Blackbox-Tests und automatisierte GUI-Tests (z.B. mit *pytest-qt*, *Selenium* oder *PyAutoGUI*) sollten bei zukünftigen Projekten fest eingeplant werden, um bessere Stabilität und Nutzerfreundlichkeit zu gewährleisten.

### **Zeitaufwändige Asset-Erstellung**

Die Suche nach geeigneten Bildern für die Gärten nahm mehr Zeit in Anspruch als geplant. Eine klarere Ressourcenplanung oder die frühzeitige Einbindung einer Bilddatenbank hätte hier Abhilfe geschaffen.

## **Wichtige Erkenntnisse und erworbene Skills**

### **Planungskompetenz:**

Das Projekt zeigte deutlich, wie wichtig eine solide Vorplanung ist.

### **Technische Fähigkeiten:**

Neue Frameworks (PyQt6, Pygame) und Tools (DALL-E, GIMP) wurden erlernt und erfolgreich eingesetzt.

### **Projekt- und Zeitmanagement:**

Es wurde anerkannt, dass klare Meilensteine und eine bessere Prioritätensetzung erforderlich sind.

### **Ausblick:**

Zukünftige Weiterentwicklungen könnten beinhalten:

- Integration von Windows-Benachrichtigungen, bspw. wenn ein Timer abgelaufen ist
- Möglichkeit Projekte zu archivieren, statt wie aktuell nur zu löschen
- Umfangreichere Statistiken zu den Projekten erstellen
- Scroll-Gärten ermöglichen, um so die Dimensionen eines Gartens zu erweitern
- Design bzw. visuelle Elemente sowohl in der GUI als auch in den Gärten überarbeiten