Plantivity

Gamified Produktivity-App for iOS

Projektbericht von Thomas Moissl & Uranik Berisha

Tag der Einreichung: 12. November 2022

Gutachter: Dr.-Ing. (habil.) Stefan Göbel
Betreuer: Wolfgang Brabänder

Darmstadt

TECHNISCHE UNIVERSITÄT DARMSTADT

Fachbereich Elektrotechnik und Informationstechnik

Serious Games

Inhaltsverzeichnis

1 Projektdefinition **4**
- 1.1 Inhalte . . . 4
  - 1.1.1 Thema . . . 4
  - 1.1.2 Beschreibung . . . 4
  - 1.1.3 Zielgruppe . . . 5
- 1.2 Konzeption . . . 6
  - 1.2.1 Konzeptionelle Herausforderung . . . 6
  - 1.2.2 Arbeitspakete der Stufen . . . 6
  - 1.2.3 Tools . . . 7

2 Projektplanung **8**
- 2.1 Framework . . . 8
- 2.2 Features . . . 10
- 2.3 Arbeitspakete . . . 12

3 Entwicklung **18**
- 3.1 Projektadministration . . . 18
  - 3.1.1 Alphaversion . . . 18
  - 3.1.2 Betaversion . . . 18
  - 3.1.3 Tests . . . 19
- 3.2 Beschreibung und Herangehensweise der Arbeitspakete der Alphaversion . . . 19
  - 3.2.1 Datenstrukturen . . . 19
  - 3.2.2 Views . . . 20
  - 3.2.3 Statistik . . . 21
  - 3.2.4 Achievements . . . 21
- 3.3 Beschreibung und Herangehensweise der Arbeitspakete der Betaversion . . . 22
  - 3.3.1 Datenstrukturen . . . 22
  - 3.3.2 Views . . . 22
  - 3.3.3 Pflanzen . . . 23
- 3.4 Beschreibung und Herangehensweise der Arbeitspakete der finalen Version . . . 24
  - 3.4.1 User Experience Verbesserungen . . . 24
  - 3.4.2 Hall of Fame Überarbeitung . . . 24
  - 3.4.3 Balancing der Pflanzen . . . 25

4 Zusammenfassung und Ausblick **26**
- 4.1 Ergebnis . . . 26
  - 4.1.1 Arbeitspakete . . . 26
  - 4.1.2 Zeitplanung . . . 26

4.2 Highlights 26
4.3 Offene Punkte und Erweiterungen 27

1 Projektdefinition

1.1 Inhalte

1.1.1 Thema

Plantivity ist eine gamified Produktivitäts-App für iOS, deren Hauptaufgabe darin besteht, durch Gamification die Motivation und so auch Produktivität des Nutzers für verschiedene Projekte zu steigern. Dabei soll sie dem Nutzer bei der Aufzeichnung und Rückverfolgung von in Aktivitäten aufgewendeter Zeit assistieren.
Erhältlich ist die App mit folgendem Link:

`https://apps.apple.com/app/plantivity/id1644563701`

Diese Version der App enthält einige Features, um die App zu überprüfen und zu bewerten. Eine Minute gibt etwa einen weitaus größere gearbeitete Zeit gutgeschrieben, sodass die Stufen der Projekte ohne Aufwand seitens der Prüfer erhöht werden können. Des Weiteren ist es möglich, im Shop auf die Coin-Anzeige zu klicken, um 100 Coins zu erhalten.

1.1.2 Beschreibung

Das Fundament der App bilden zwei wichtige Datenstrukturen, welche die Aufzeichnungen der produktiven Zeit übernehmen. Die Aufzeichnungen werden in sogenannten *Projects* gruppiert, welche entweder aus weiteren *Projects* bestehen können, um das Erstellen von Hierarchien zu ermöglichen, oder aus *Sessions*, welche zeitlich beschränkte Arbeitseinheiten an dem angegebenen Projekt bilden. Diese werden gespeichert und für Features der Gamification, aber auch spätere Betrachtung von Produktivitätsstatistiken aufbereitet. Die grundlegende Funktionalität wird umgesetzt, indem für diese Datenstrukturen CRUD-Operationen verfügbar sind. Der Nutzer kann somit selbstständig verschiedene Projekte anlegen und für diese Sessions mit festgelegter Zeit starten, in welcher der Nutzer am vorgesehenen Projekt arbeiten soll.
Die Gamification wird durch ein System erzeugt, indem der Nutzer sich um virtuelle Pflanzen kümmern muss. Dafür wird für jedes angefangene Projekt jeweils eine neue Pflanze gepflanzt, welche durch regelmäßiges Abschließen von Sessions gepflegt wird. Dabei gibt es verschiedene Pflanzen mit verschiedenen Stufen, welche durch einen entsprechenden Arbeitsaufwand am Projekt verbessert werden können. Sollte ein Projekt vernachlässigt werden, bildet sich die Pflanze zurück und verkümmert schließlich.

Eine virtuelle Monetarisierung der Pflanzenstufen durch eine virtuelle Währung soll die Nutzer motivieren an selbstbestimmten Projekten zu arbeiten, um Coins zu verdienen, aber auch, um durch regelmäßige Arbeit an allen Projekten das Zurückbilden der Pflanzen zu verhindern.



Abbildung 1.1: Illustrationen von vectorjuice / Freepik

Mit diesen Coins können über einen Shop neue Pflanzen bzw. deren Samen zum Erstellen von Projekten gekauft werden. Ebenfalls werden Hilfsmittel angeboten, welche die spielerischen Aspekte steigern oder für ernstere Nutzer eine Option liefern, die App nahezu ungehindert zu nutzen.
Des Weiteren werden Statistiken erstellt, welche den Fortschritt mit mehr Details als nur die Stufen der Pflanzen aufzeichnen. Hierbei gibt es allgemeine Statistiken, welche visuell vermitteln, wie der Nutzer seine Zeit verwendet. Für interessierte Nutzer sollen die exakten Zahlen jedoch ebenfalls verfügbar sein.
Als weitere Ebene der Motivation und Gamification gibt es Achievements, durch welche der Nutzer über Meilensteine in seinem Fortschritt informiert wird, aber auch angeregt wird, auf diese hinzuarbeiten. Durch zusätzliche regelmäßige Herausforderungen wird dies auch in kleinerem Maß erreicht. Achievements und regelmäßige Herausforderungen sind des Weiteren als ein Belohnungssystem vorgesehen, um den Nutzer nicht nur dazu zu motivieren, möglichst alle Funktionen der App zu probieren, sondern auch neue Dinge für sich zu testen.
Zuletzt soll der Fortschritt über eine View leicht teilbar sein. Dabei sind die höchsten, bereits erreichten Stufen jeder Pflanze sichtbar. Diese Funktion gibt einerseits die Möglichkeit, kompetitiv zu sein, aber auch einen Punkt, an welchem Nutzer sich über die App im Allgemeinen austauschen können.

1.1.3 Zielgruppe

Plantivity richtet sich an Nutzer, die Projekte haben, welche durch das zusätzliche Erfolgserlebnis einer spielerischen App profitieren können. Dabei wird angenommen, dass die Nutzer die Regeln einer solchen App ernst genug nehmen können, bzw. sich ehrlich verhalten und daran interessiert sind, ihre Produktivität zu

steigern. Es gibt meist Nutzer, welche diese Annahme nicht erfüllen, wenn diese etwa nur die Achievements oder höchsten Stufen erreicht wollen, ohne die tatsächliche Funktionalität der App wahrzunehmen. Dagegen können kaum Maßnahmen ergriffen werden. Diese Nutzergruppe wird deshalb aus der Zielgruppe ausgenommen, um eine mögliche Minderung der Qualität für den Rest der Nutzer zu vermeiden. Neben dem Schwerpunkt des Pflegens von Pflanzen wird auch durch Meilensteine, Herausforderungen und Statistiken möglichst vielen Nutzern dabei geholfen, mehr Motivation für ein Projekt aufbringen zu können. Dadurch werden Nutzer angesprochen, welche auch in Spielen auf vergleichbare Belohnungssysteme positiv reagieren. Durch die verschiedenen Stufen und Arten der Pflanzen ist es vorgesehen, dass sich die App mit der Zeit von selbst an den Aufwand anpasst, welchen der Nutzer in seine Projekte investiert. Somit ist die Nutzung der App unabhängig von der Größe der Projekte, wodurch die Zielgruppe unabhängig vom Alter, der verfügbaren Freizeit und den individuellen Umständen des Nutzers ist.

1.2 Konzeption

1.2.1 Konzeptionelle Herausforderung

Um den praktischen Aspekt des Spiels zu gewährleisten, wird ein passendes Belohnungssystem entwickelt, welches durch eine implizite Motivation den Nutzer anregt, produktiv zu sein und die App zu nutzen. Dabei müssen sowohl die Ziele aus der Entwickler-Perspektive als auch aus der Endnutzer-Perspektive erfüllt werden. Deshalb soll der Nutzer bewogen werden, den organisatorischen Aufwand des Trackings auf sich nehmen zu wollen. Gleichzeitig soll die Verwendung der App den Nutzer dazu motivieren, produktiv zu sein, indem er sich Ziele setzt und diese aufgrund der spielerischen Aspekte einhalten will.

Eine besondere Herausforderung bildet dabei das Balancieren widersprüchlicher Ziele:

- Motivation und Vermeidung von Wertlosigkeit kleiner Pflanzen/Projekte im späteren Verlauf
- Zu viele Optionen, sodass von Produktivität abgelenkt wird und zu wenig Optionen, sodass der Gamification Aspekt nicht genug vorhanden ist
- Nutzer sollen an bestehenden Projekten weiterarbeiten wollen aber gleichzeitig versuchen, bestehende Projekte abzuschließen

1.2.2 Arbeitspakete der Stufen

1. Prototyp (Datenmodelle und grundlegende Funktionalitäten)
    a) CRUD Operationen für Projekte
    b) Absolvieren von Sessions
    c) Zusammenfügen von Projekten
    d) Kategorisierung von Projekten
2. Alpha Version (Erweiterte Funktionalität und grundlegende Gamification)
    a) Darstellung der Pflanzen

b) Ausgabe der Daten in Statistiken

c) Steigerung der Motivation durch Achievements

d) Speicherung persönlicher Rekorde

3. Beta-Version (Erweiterte Gamification)

a) Stufensystem der Pflanzen

b) Einführung einer Währung mit Coins

c) Hinzufügen des Shops

d) Regelmäßige Herausforderungen

4. Release

a) Implementierung von Features für zusätzliche Steigerung der Nutzerfreundlichkeit

b) Beseitigung der Bugs, welche in Alpha- und Beta-Phase aufgefallen sind

Die Planung für das Vorgehen der Entwicklung sieht für jede Phase das Hinzufügen neuer Funktionalitäten so vor, dass es nach jeder Phase eine abgeschlossene App entsteht, welche in der darauffolgenden Phase erweitert wird. Die Entwicklung wird dafür in vier Phasen aufgeteilt, sodass zu jedem Zeitpunkt eine Version existiert, welche für den Release bereit ist. Das ist damit zu begründen, dass die Zielsetzung für den festgelegten Zeitrahmen ambitioniert ist und potenziell nicht vollständig umgesetzt werden kann. Da allerdings für den Abschluss des Projekts der Release von Plantivity im iOS App Store vorgesehen ist, ist dieser Schritt sinnvoll.

1.2.3 Tools

Für die Umsetzung werden folgende Tools und Frameworks verwendet:

Als IDE für das Projekt wird die Apple eigene Entwicklungsumgebung Xcode in der Version 13.3.1 verwendet. Dadurch können für die Tests während der Entwicklungszeit unter anderem die eingebundenen Simulatoren genutzt werden.

Für die Implementierung der Logik wird die aktuelle Version 5.5 der Programmiersprache Swift verwendet. Dadurch kann als GUI-Frameworks SwiftUI verwendet werden, welche für eine vereinfachte Entwicklung durch deklarative Erstellung der Views erstellt wurde.

Zur Umsetzung der Anforderungen an die Speicherung der Daten ist eine Datenbank unabdingbar. Für das Speichern von Daten bietet Apple ein Persistance-Framework an, welches den geregelten Zugriff erlaubt. Da Core Data nach dem Entity-Relationship-Modell, lassen sich so einfach Relationen umsetzen, welche für die Modellierung der verwendeten Datenstruktur sinnvoll sind.

Da die Assets der Pflanzen, Items und Achievements selbst erstellt werden sollen, wird die iOS-App Procreate verwendet. Das erleichtert etwa das Exportieren der Assets im richtigen Dateiformat und passenden Größen.

Vorgesehen ist als Ergebnis, dass die beschriebene App vollständig umgesetzt und im iOS App Store veröffentlicht ist.

2 Projektplanung

2.1 Framework

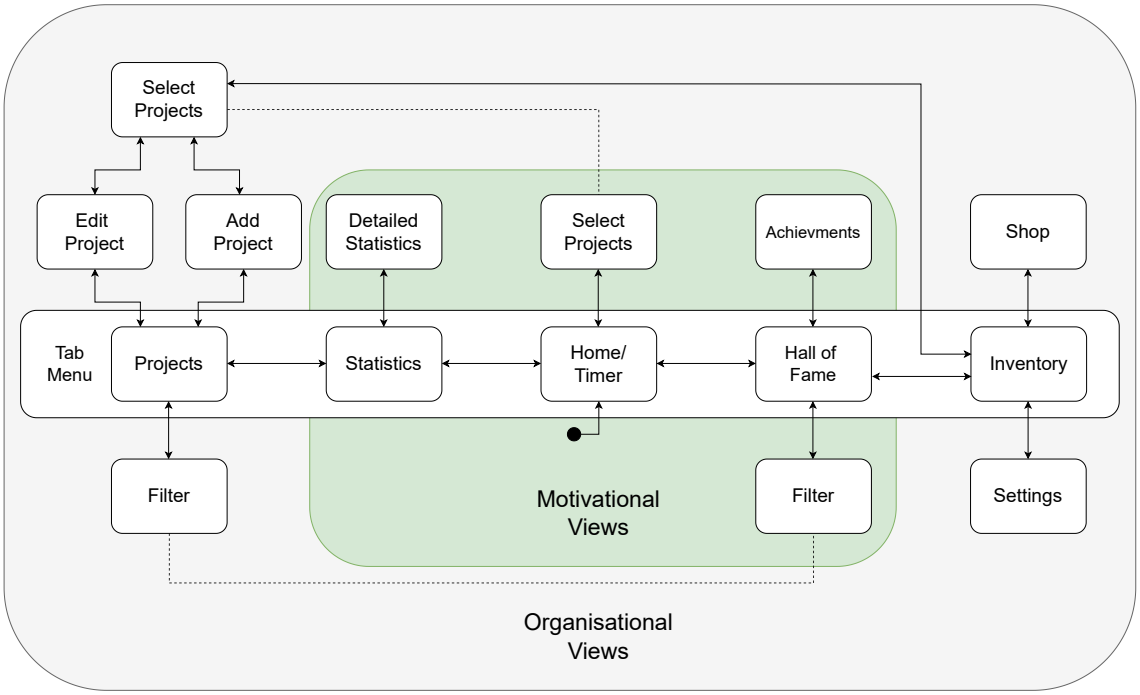


Abbildung 2.1: Geplanter User Flowchart

Die App stellt voraussichtlich fünf Tabs zur Verfügung, welche über eine Tabbar am unteren Rand des Bildschirms navigiert werden können. Für die Flowchart der App siehe 2.1. Deren funktionaler Zusammenhang ist in 2.2 abgebildet. Für ein Mockup der über die Tabbar erreichbaren Views, siehe 2.3. Die Positionen sind so angeordnet, dass die wichtigsten Views für die Motivation zentral und damit leicht zugänglich sind. Weitere Tabs zur Organisation von Projekten und Anpassung der Einstellungen der App sind hingegen wegen ihrer selteneren Benutzung am Rand vorzufinden.
Der relevanteste Tab ist die *Timerview*, in welcher ein Projekt ausgewählt werden kann, für welches anschließend ein Timer gestartet wird. Es soll außerdem möglich sein, ein Item aus dem Inventory zu wählen, welches einen Effekt auf die Session hat, wie beispielsweise eine erhöhte Belohnung, die wiederum für neue Items verwendet werden kann. Ist noch kein Projekt angelegt, soll ein Nutzer direkt die Option erhalten, dies zu tun, ohne erst zur Projektübersicht wechseln zu müssen. Läuft der Timer ab, ist eine Session für das gewählte Projekt abgeschlossen. Anschließend werden die Statistiken und gegebenenfalls die Hall of Fame angepasst. In diesem Abschnitt sind alle Funktionen Must-Have, da sie den Kern der Plantivity-App darstellen und demnach

unerlässlich sind für die weiteren Funktionen.

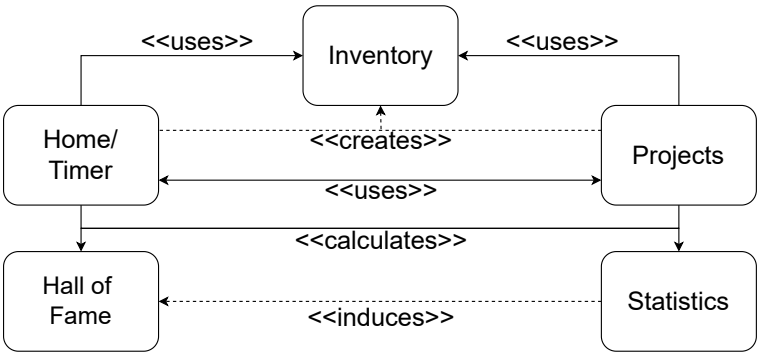


Abbildung 2.2: Geplantes Komponentendiagramm

Die nächste View, welcher der Motivation dient, behandelt die über die Produktivität des Nutzers erstellten Statistiken. Dabei soll auf den ersten Blick durch visuelle Aufarbeitung der Daten ein Überblick über die in der *Timerview* absolvierten Sessions und die Projekte des Nutzers gegeben werden. Dennoch wird in einer weiteren View eine ausführlichere Statistik zu finden sein, welche konkrete Zahlen liefert. Statistiken sind als Nice-to-Have einzustufen, da sie neben der Motivation keine für die Funktion der App unerlässliche Aufgabe erfüllen.
Die *HallOfFameView*, in welcher besondere Projekte betrachtet werden können, ist die letzte Motivational View. Sie motiviert durch die Erinnerung an bisherige abgeschlossene oder auch gescheiterte Projekte. Zusätzlich sind von hier aus auch Achievements, bzw. Meilensteine einsehbar, welche im Verlauf der Nutzung der App durch das Absolvieren von Sessions freigeschaltet werden können, indem bestimmte Werte in den Statistiken erreicht werden. Die Umsetzung von Plantivity als gamified Produktivitäts-App ist ohne Achievements nicht denkbar, da sie sehr hohe Relevanz für gesteigerte Motivation haben. Damit sind die Features unerlässlich und ein Must-Have für Plantivity.
Die erste organisatorische View ist das Projekte-Tab. Hier können neue Projekte angelegt bzw. die entsprechende Pflanze gepflanzt werden, aber auch vorhandene Projekte verwaltet oder abgeschlossen werden. In dieser View sind ebenfalls Items aus dem Inventory verwendbar. Da ohne die Funktionalitäten dieser View die restliche App nicht verwendet werden kann, ist sie ebenfalls als Must-Have klassifiziert.
Der letzte Tab befasst sich mit der Verwaltung der Gegenstände des Nutzers. Über dieses Tab soll auch der Shop verfügbar sein, in welchem neue Gegenstände erworben werden können, welche nach dem Kauf im Inventory vorzufinden sind. Zuletzt sind hier auch die Einstellungen der App vorzufinden. Dabei ist der Shop und das Inventar ein Must-Have, da der Gamification-Aspekt stark auf einem internen Währungssystem beruhen soll, welches ohne Shop keinen Wert für den Nutzer haben kann. Einstellungen für die App ergeben sich erst im Laufe der Entwicklung, vorweg sind diese als Nice-to-Have einzustufen, was sich abhängig von den implementierten Features noch ändern kann.

2.2 Features



Abbildung 2.3: Mockup Views

1. **Tracking von Projekten**
   Plantivity hilft Nutzern vor allem dabei, den Überblick über den in Projekte investierten Aufwand zu behalten. Deshalb ist das erste Feature das Tracking von Projekten. Hierfür kann der Nutzer Projekte anlegen und dann die Zeit, welche mit diesen Projekten verbracht wird, mit Hilfe von Sessions aufgezeichnet werden. Es ist auch möglich, dass ein Projekt untergeordnete Projekte besitzt. Das kann dazu beitragen, dass das Tracking detaillierter und damit potentiell hilfreicher für den Nutzer ist. Der Tracking-Aspekt wird vor allem in den Statistiken deutlich, in welchen der Nutzer die über lange Zeit regelmäßig eingetragenen Einheiten aufgearbeitet präsentiert bekommt, um so den Fortschritt nachverfolgen zu können. Tracking ist kritisch für die App, da es neben der Gamification das Hauptfeature der App darstellt und somit den Sinn von Plantivity repräsentiert.

2. **Visuelle Darstellung des aufgebrachten Aufwands**
   Die zuvor genannte Darstellung des Aufwands ist dementsprechend ebenfalls relevant für die Motivation des Nutzers, allerdings ist der Umfang wie bereits in der Zusammenfassung genannt abhängig von der Zeit, welche dafür verfügbar ist. Das ist vor allem in der Hinsicht relevant, dass aufgrund mangelnder Erfahrung noch nicht gut abgeschätzt werden kann, wie viel Zeit dieses Arbeitspaket in Anspruch nimmt, worauf allerdings später eingegangen werden soll. Es ist dennoch vorgesehen, dass der Nutzer in den Statistiken die Option hat, diese individuell anzupassen, um möglichst genau die gewünschten Daten zu sehen. Die grafische Darstellung steht für diese anpassbaren Statistiken im Vordergrund. Falls der Nutzer allerdings ausführlichere Informationen erhalten möchte, sollen hiervon ebenfalls alle gesammelten verfügbar sein.

   Für die Funktionalität der App sind die Statistiken nicht kritisch, dienen aber als Verbindungsstück zwischen der datenfokussierten Grundfunktionalität der App selbst und der Gamification. So kann der Nutzer aus den Statistiken die Zusammenhänge erschließen, falls das Interesse besteht. Die im nächsten Schritt behandelte Gamification soll allerdings ebenfalls einen Überblick über den Fortschritt liefert. Die Statistiken erlauben jedoch einen tieferen Einblick.

3. **Gamification des Trackings**
   Der Gamification Aspekt umfasst mehrere Unterfeatures. Das erste und wichtigste ist die Verbindung von Projekten und Pflanzen. Beim Erstellen eines Projekts soll ein Samen für eine Pflanze gepflanzt werden, welcher aufgeht und entsprechend des Aufwands, welchen der Nutzer in das Projekt investiert, wachsen, oder bei zu wenig Arbeit auch eingehen. Darüber hinaus, dass ein Nutzer versuchen soll, eine möglichst große Pflanze zu erzeugen, bzw. seine Pflanze nicht sterben zu lassen, geben diese auch eine Belohnung nach jeder Session. Die Größe dieser Belohnung wird durch ein zugehöriges Stufensystem festgelegt. Dabei bedeutet eine höhere Stufe, dass die Pflanze mehr Pflege benötigt aber dafür auch größere Belohnungen ausschüttet. Aus einer niedrigeren Stufe folgt, dass die Pflanze auch mit wenig Arbeit nicht eingeht und entsprechen auch nur eine kleine Belohnung gibt, sollte eine Session abgeschlossen werden. Die Stufe einer Pflanze wird über einen bestimmten Zeitrahmen durch den in das Projekt investierten Aufwand ermittelt und kann sich somit im Laufe der Zeit verändern. Verschiedene Pflanzen verhalten sich hinsichtlich der Stufen auch unterschiedlich, sodass es beispielsweise eine Pflanze gibt, welche für die kleinste Art von Projekten geeignet ist, da sie nicht eingehen kann. Entsprechend niedrig fallen allerdings die Belohnungen im Vergleich zu anderen Pflanzen selber Stufe aus.

   Die zuvor genannte Belohnung ist voraussichtlich in Form von Coins, einer in-game Währung, von welcher sich der Nutzer Gegenstände in einem Shop kaufen kann, um seine Pflanzen besser zu pflegen oder auch gegen Vernachlässigung bis zu einem gewissen Grad resistent machen kann. Das umfasst beispielsweise Samen für neue Projekte, Dünger, welcher die Werte einer Pflanze steigert oder Gegenstände zum Pausieren/Einfrieren von Projekten. Plantivity ermöglicht es somit, Nutzern, welche sich regelmäßig um ihre Projekte kümmern, auch Auszeiten zu nehmen. Da durchgehende hohe Leistung auf längere Zeit nicht im Sinne von Produktivität sein kann, muss diese Möglichkeit bestehen. Allerdings stellt der Preis der Gegenstände sicher, dass das Vergleichen von Fortschritt zwischen Nutzer aussagekräftig bleibt. Durch den Shop wird auch ein Gefühl von Wert der Pflanzen vermittelt, da diese die Hauptquelle für Coins sind und nur damit Gegenstände im Shop erworben werden können. Das führt dazu, dass der Nutzer motivierter ist, seine Pflanzen nicht zu vernachlässigen, was wiederum zu gesteigerter Leistung an korrespondierenden Projekten selbst führt.

   Da Plantivity vor allem durch diese Gamification motivieren soll, ist nicht nur das Feature selbst unerlässlich, sondern auch die Qualität der Umsetzung kritisch. Dementsprechend wird die Gamification auf so vielen Ebenen wie möglich angewendet, da manche Nutzer eher durch Statistiken, andere aber durch das Belohnungssystem im Spiel angeregt werden können.

4. **Achievements zur Feststellung von Langzeitforschritten**
   Ein weiterer, bei vielen Nutzern beliebter Punkt der Gamification liegt außerdem in Achievements. Diese geben ebenfalls bis zu einem gewissen Grad einen Überblick über die vom Nutzer erbrachten Leistungen und vereinfachen das Vergleichen der Spielstände zwischen Nutzern. Somit führen sie zu einer Erweiterung der Zielgruppe, da in Spielen viele durch diese besonderen Herausforderungen angetrieben werden.

   Achievements sind allerdings unausweichlich beschränkt, indem sie für manche Nutzer zu anspruchsvoll und für andere zu einfach sind. Das trifft auch auf Plantivity zu, da die genaue Nutzung der App von jedem individuellen Nutzer unbekannt ist. Diese könnte von einer Session pro Woche zu einer vollen Arbeitswoche reichen und selbst diese Grenzen sind nicht garantiert. Somit muss abgewogen sein, welche Nutzergruppe eine gute Herausforderung durch die Achievements bekommen soll und welche Zielgruppe manche Achievements sofort oder alternativ nie erreichen wird.

Diesen Effekt abzuschwächen ist möglich, indem die Achievements anhand ihrer Schwierigkeitsstufe klassifiziert werden oder für eine Art von Achievement mehrere Zahlenwerte zu nehmen, sodass dem Nutzer die entsprechende Schwierigkeit durch den selbst aufgebrachten Aufwand vermittelt wird. Die genaue Umsetzung des Features zeigt sich deshalb erst während der Entwicklung. Fest steht allerdings bereits an dieser Stelle, dass Achievements kritisch für die App sind.

5. **Hall of Fame für die Darstellung abgeschlossener Projekte**
   Um diese Problematik weiter abzuschwächen, wird der Nutzer zusätzlich durch persönliche Bestleistungen im Hall of Fame seiner Pflanzen ebenfalls motiviert und angespornt, diese immer wieder aufs Neue zu übertreffen. Hierfür wird eine View zur Verfügung gestellt, in welchen die Pflanzen, welche zu den Projekten der bestimmten Bestleistungen gehören, zusammen mit dem Rekord selbst angezeigt werden. Dadurch wird eine ähnliche Funktion wie von Achievements umgesetzt, indem der Nutzer eine Ansicht hat, welche seine Leistungen repräsentiert.

   Achievements sind dennoch eine bereits stärker mit Spielen verbundene Funktion und deshalb bereits bekannter. Die Kombination beider Features ist also wichtig und wie auch beim Belohnungssystem gilt, dass die Qualität der Umsetzung kritisch für die Wirkung der App ist, besonders für den Bereich der Zielgruppe, welcher durch diesen Aspekt angesprochen wird.

6. **Timer zur Steigerung der Motivation in einzelnen Sessions**
   Die Aufzeichnung einzelner Sessions erfolgt, indem ein Timer für das Projekt gestartet wird. Während dieser abläuft, ist vorgesehen, dass der Nutzer einer Tätigkeit nachgeht, welche zu Fortschritt im entsprechenden Projekt führt. Eine Session kann nicht pausiert oder vorzeitig beendet, sondern nur abgebrochen werden, damit der Nutzer bei seinen zuvor gesetzten Zielen bleibt und die vorgesehene Zeit produktiv ist. Dies vermeidet nicht nur vorzeitiges Beenden von produktiven Einheiten, sondern fördert auch die Fähigkeit, Zeiteinheiten korrekt vorzuplanen. Damit steigert die App nicht nur durch langfristige Motivation die Produktivität, sondern hilft dem Nutzer, während er an seinem Projekt arbeitet, fokussiert zu bleiben. Es ist anzumerken, dass das Handy oftmals eine Ablenkung darstellen kann. Wenn ein Nutzer zu Beginn seiner Arbeit eine Session starten möchte und die App öffnet, wird das Erste, was er auf seinem Handy sieht, auch der Timer sein, was App-interne Ablenkung vermeiden soll. Der Timer ist deshalb für den Aspekt der Motivation bereits sehr wichtig, aber zum Aufzeichnen von Sessions kritisch, da ohne diesen kein Tracking möglich ist.

2.3 Arbeitspakete

Die Arbeitspakete werden im Folgenden ohne Zuständigkeit beschrieben, da die Entwicklung in Pair-Programming erfolgt.

1. Datenstrukturen
   Für die effiziente Speicherung aller Daten müssen zuerst einige Datenstrukturen erstellt werden. Dass die unter diesen Punkt fallenden Arbeitspakete relevant sind, folgt daraus, dass eine der technisch anspruchsvollsten Aufgaben der App die Speicherung und Verwendung der gesammelten Daten ist. Eine weitere Herausforderung liegt darin, die Datenstruktur so anzulegen, dass sie möglichst alle im Laufe des Projekts neu hinzukommenden Anforderungen bereits erfüllt oder alternativ flexibel genug ist, um

entsprechend angepasst zu werden, ohne vorhandene Arbeit abändern zu müssen.
Verwendet werden soll hierbei Core Data, was genau diese leichte Erweiterung der Datenstrukturen erlaubt.

a) Projekte
   Für ein Projekt müssen einige Daten abgespeichert werden. Dazu gehören etwa Name und Kategorie des Projekts, aber auch die Beziehungen zu anderen Objekten wie den Sessions, welche für das Projekt abgeschlossen werden und auch die Pflanze, die aus Sicht des Nutzers das Projekt repräsentiert. Es ist außerdem geplant, dass ein Projekt untergeordnete (und dementsprechend auch übergeordnete) Projekte haben kann, um feingliedrigere Bearbeitung und Tracking zu erlauben.

b) Sessions
   Sessions sind für den Nutzer nicht direkt einsehbar. Sie werden erstellt, wenn der Timer für eine Arbeitseinheit für ein Projekt abläuft und sind vor allem für detaillierte Statistiken relevant. Dementsprechend werden auch hier während der Entwicklung weitere Attribute hinzugefügt.

c) Pflanzen
   Für Pflanzen ist eine weitere Datenstruktur anzulegen, da die verschiedenen Werte jeder Stufe, welche für jede Pflanze individuell sein sollen, aber auch Preis abgespeichert werden müssen. Es ist ebenfalls möglich, dass Pflanzen zusätzliche Werte haben, welche bestimmtes positives Verhalten oder auch das Auslassen von entsprechend negativem Verhalten des Nutzers bezüglich seiner Projekte durch erhöhte Belohnungen fördern können. Diese Erweiterung ist allerdings nur dann vorgesehen, falls noch Zeit übrig ist.

d) Items
   Die letzte Datenstruktur ist für die Items des Shops. Da diese zwar unerlässlich für die Funktionalität des Shops sind, aber im Verhältnis zu beispielsweise Sessions sehr selten gebraucht werden und auch keine Beziehungen zu anderen Datenstrukturen haben, ist eine schnellere Umsetzung zu erwwarten.

2. Views (siehe Abb. 2.3)

a) Timer
   Die erste View, welche der Nutzer nach dem Öffnen der App sieht, ist die *Timerview*. Da dort auch der namensgebende Timer während einer Session abläuft, wird auf diesem Tab auch die meiste Zeit verbracht, weshalb es die höchste Relevanz hat. Enthalten ist außerdem eine Subview zur Projektauswahl für die Session. Diese kann auch an anderen Stellen in der App wiederverwertet werden, wird aber als Teil dieses Arbeitspakts erstellt. Der Fokus wird auf einer ansehnlichen Umsetzung des Timers liegen.

b) Statistiken
   Für die Anzeige der Statistiken ist auch eine eigene View vorgesehen. In dieser soll auf den ersten Blick ersichtlich sein, wie produktiv der Nutzer im aktuellen Zeitraum ist und in welchen Kategorien der größte zeitliche Aufwand liegt. Dabei sollen die nötigen Informationen aus den gespeicherten Sessions und Projekten verarbeitet werden. Die beiden Herausforderungen sind dabei, dass die Daten verständlich, aber auch unabhängig von der insgesamt eingetragenen Produktivität des Nutzers angezeigt werden. Hierfür bietet sich eine prozentuale Darstellung an.
   Dennoch ist es für eine Trackingapp nötig, dass Nutzer die Einzelheiten über ihre Daten abrufen können. Dafür ist eine Subview vorgesehen, in welcher eine größere Menge an Informationen angezeigt wird. Dabei wird auch auf die visuelle Aufarbeitung und die prozentuale Darstellung

verzichtet und die objektiven Zahlenwerte gezeigt.

c) Achievements
Die letzte Motivational View ist für die Hall of Fame der Pflanzen. In dieser wird versucht, durch die Darstellung der Pflanzen abgeschlossener Projekte den Nutzer dazu zu motivieren, weitere Projekte abzuschließen, um diese View aufzuwerten. Gleichzeitig wird so eine Übersicht über die Projekte geboten, welche es erlaubt, die projektspezifischen Informationen einzusehen.
Von diesem Tab aus sind auch die Achievements erreichbar. Diese werden im Lauf der Benutzung der App freigeschaltet und können dort als Meilensteine bzw. Beweis für bestimmte Leistungen eingesehen werden.

d) Projekte
Die erste organisatorische View ist für die Verwaltung von Projekten. Diese sollen in der View hierarchisch aufgelistet sein, sodass eine möglichst angenehme Navigation möglich ist. Die Sortierung nach Status des Projekts ist ebenfalls sinnvoll, damit abgeschlossene Projekte beispielsweise am Ende der Liste vorzufinden sind.
Des Weiteren müssen auch die Optionen für die Projekte und die dafür benötigten Subviews umgesetzt werden. Die erste ist hierbei eine Subview zum Erstellen von Projekten, welche von der Projektansicht aus aufgerufen werden kann. Darin wird beispielsweise der Name, die Kategorie und falls nötig das übergeordnete Projekt angegeben.Die Umsetzung der Projekte ist nötig, damit Projekte für Statistiken unterschieden werden können und auch um für den Nutzer die Projekte individueller gestalten zu können.
Ebenfalls ist eine Subview zum Bearbeiten von Projekten notwendig, für welche eine leicht abgewandelte Version der vorher behandelten Subview sein kann, da sich die meisten Punkte, welche bei dem Erstellen angegeben werden können und welche im Nachhinein bearbeitet werden können, überschneiden. Es soll allerdings die Option zur Verfügung gestellt werden, Projekte zusammenzufügen. Das führt dazu, dass alle Sessions von einem Projekt auf ein anderes Übertragen werden und das Projekt anschließend gelöscht wird. Diese beiden Views umfassen allerdings auch die CRUD-Operationen für die Projekte, welche umgesetzt werden müssen.

e) Shop
Um dem in späteren Unterkapiteln folgende Währungssystem Sinn zu verleihen, sind Items vorgesehen, welche in diesem Tab verwaltet werden können. In einer Subview ist außerdem ein Shop vorgesehen. In diesem können verschiedene Items erworben werden, welche in der Shop View zu sehen sind.
In einer weiteren Subview sind die Einstellungen der App zu finden. Dabei ist noch keine Zeit eingeplant, da sich der Umfang erst im Laufe der Entwicklung zeigen wird, aber voraussichtlich vernachlässigbar ist.

3. Konzepte
Neben der Funktionalität und den Views muss auch Zeit investiert werden, um den Gamification-Aspekt der App so zu entwickeln, dass eine möglichst große Gruppe von Nutzern dadurch den erwünschten Effekt der Produktivitätssteigerung erhält. Im Folgenden werden diese genannt.

a) Stufensystem der Pflanzen
Wie in vorherigen Kapiteln angesprochen, wird die Anpassung des Anspruchs der App an den Aufwand eines Nutzers durch ein Stufensystem der Pflanzen umgesetzt. Hierbei wird eine Stufe durch die in Spielen üblichen Klassen S, A, B, C, D repräsentiert. Der Erhalt einer höheren Stufe

erfordert mehr regelmäßige Arbeitseinheiten für das Projekt. Als Gegenleistung belohnt eine Pflanze höherer Stufe auch mit einer größeren Anzahl an Coins, welche im nächsten Abschnitt behandelt werden. Da es auch verschiedene Pflanzen gibt, kann es sinnvoll sein, für diese verschiedene Werte pro Stufe zu verteilen, sodass ein Unterschied über den visuellen Aspekt hinaus entsteht.

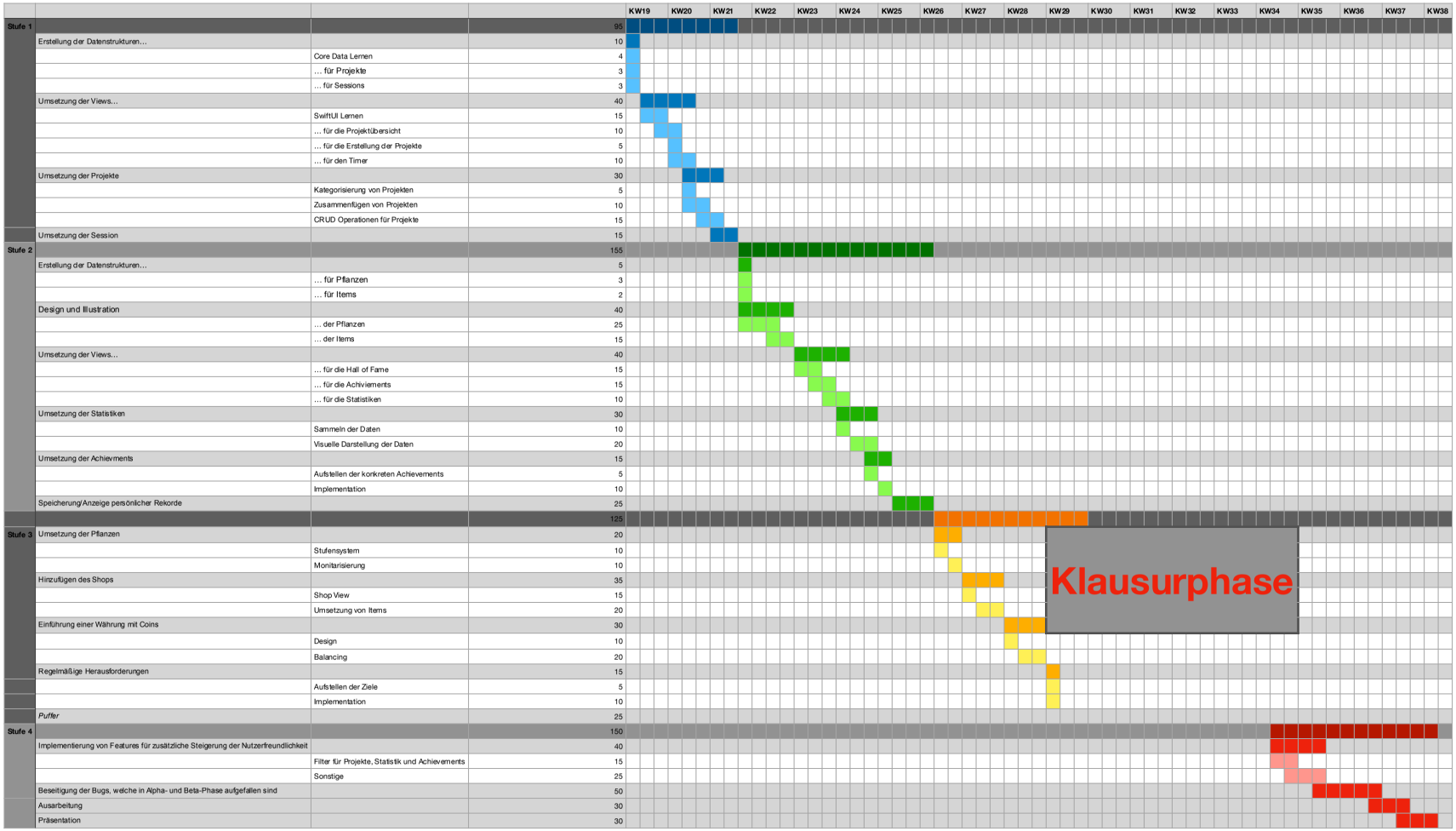
b) Währungssystem der Coins
Die Währung, welche jeden Wert innerhalb der App bestimmt sind Coins. Sie werden durch das Absolvieren von Sessions, möglicherweise aber auch durch das Erreichen eines Achievements verdient. Ausgegeben kann der Nutzer sie im Shop für Items, wie etwa Samen für besondere Pflanzen oder zum Erhalt bestimmter Boni. Dadurch soll das Gefühl von Wert vermittelt werden, um für die Arbeit an einem Projekt ein zusätzliches Erfolgserlebnis auszulösen. Die Belohnungen und Preise müssen allerdings balanciert werden, was zusätzlich viele Tests erfordert und fehlende Erfahrung mit Gamedesign selbst erschwert die Aufgabe weiter.

4. Design
Einer der letzten Punkte ist das Design der Elemente des Spiels, welche für die Gamification unerlässlich sind. Allerdings ist keine Erfahrung bei Design vorhanden, sodass in der entsprechenden Phase ein Puffer vorhanden ist, um eventuelle Verschätzungen auszugleichen.

a) Pflanzen
Zuerst ist das wichtigste Element von Plantivity zu nennen. Das Design der Pflanzen selbst hat höchste Priorität in dieser Kategorie, da das Element der Pflanzen den Sinn hat, den Nutzer davon zu überzeugen, dem Projekt eine Relevanz zuzusprechen, welche über das Projekt selbst hinausgeht, sodass wenn die Motivation gering ist, am Projekt selbst weiterzuarbeiten, dennoch die Pflanze instand gehalten werden soll.

b) Coins
Die Währung selbst muss ebenfalls möglichst wiedererkennbar sein und auch den nötigen Wert vermitteln. An verschiedenen Stellen der App kommen Coins zum Einsatz und müssen deshalb entsprechend entworfen werden.

	Aufgabe	Unteraufgaben	Wichtigkeit	Aufwand (in h)
Stufe 1				95
	Erstellung der Datenstrukturen…		Must-Have	10
		Core Data Lernen	-	4
		… für Projekte	Must-Have	3
		… für Sessions	Must-Have	3
	Umsetzung der Views…		Must-Have	40
		SwiftUI Lernen	-	15
		… für die Projektübersicht	Must-Have	10
		… für die Erstellung der Projekte	Must-Have	5
		… für den Timer	Must-Have	10
	Umsetzung der Projekte			30
		Kategorisierung von Projekten	Nice-To-Have	5
		Zusammenfügen von Projekten	Nice-To-Have	10
		CRUD Operationen für Projekte	Must-Have	15
	Umsetzung der Session			15
Stufe 2				155
	Erstellung der Datenstrukturen…		Must-Have	5
		… für Pflanzen	Must-Have	3
		… für Items	Must-Have	2
	Design und Illustration			40
		… der Pflanzen	Must-Have	25
		… der Items	Nice-To-Have	15
	Umsetzung der Views…			40
		… für die Hall of Fame	Must-Have	15
		… für die Achiviements	Must-Have	15
		… für die Statistiken	Nice-To-Have	10
	Umsetzung der Statistiken		Nive-To-Have	30
		Sammeln der Daten	Nice-To-Have	10
		Visuelle Darstellung der Daten	Nice-To-Have	20
	Umsetzung der Achievments		Must-Have	15
		Aufstellen der konkreten Achievements	Must-Have	5
		Implementation	Must-Have	10
	Speicherung/Anzeige persönlicher Rekorde		Nive-To-Have	25
				125
Stufe 3	Umsetzung der Pflanzen		Must-Have	20
		Stufensystem	Must-Have	10
		Monitarisierung	Must-Have	10
	Hinzufügen des Shops			35
		Shop View	Must-Have	15
		Umsetzung von Items	Nice-To-Have	20
	Einführung einer Währung mit Coins			30
		Design	Nice-To-Have	10
		Balancing	Must-Have	20
	Regelmäßige Herausforderungen		Nice-To-Have	15
		Aufstellen der Ziele	Nice-To-Have	5
		Implementation	Nice-To-Have	10
	Puffer			25
Stufe 4				150
	Implementierung von Features für zusätzliche Steigerung der Nutzerfreundlichkeit		Nice-To-Have	40
		Filter für Projekte, Statistik und Achievements	Nice-To-Have	15
		Sonstige	Nice-To-Have	25
	Beseitigung der Bugs, welche in Alpha- und Beta-Phase aufgefallen sind		Must-Have	50
	Ausarbeitung			30
	Präsentation			30



Stufe 1			95
	Erstellung der Datenstrukturen…		10
		Core Data Lernen	4
		… für Projekte	3
		… für Sessions	3
	Umsetzung der Views…		40
		SwiftUI Lernen	15
		… für die Projektübersicht	10
		… für die Erstellung der Projekte	5
		… für den Timer	10
	Umsetzung der Projekte		30
		Kategorisierung von Projekten	5
		Zusammenfügen von Projekten	10
		CRUD Operationen für Projekte	15
	Umsetzung der Session		15
Stufe 2			155
	Erstellung der Datenstrukturen…		5
		… für Pflanzen	3
		… für Items	2
	Design und Illustration		40
		… der Pflanzen	25
		… der Items	15
	Umsetzung der Views…		40
		… für die Hall of Fame	15
		… für die Achivements	15
		… für die Statistiken	10
	Umsetzung der Statistiken		30
		Sammeln der Daten	10
		Visuelle Darstellung der Daten	20
	Umsetzung der Achievments		15
		Aufstellen der konkreten Achievements	5
		Implementation	10
	Speicherung/Anzeige persönlicher Rekorde		25
			125
Stufe 3	Umsetzung der Pflanzen		20
		Stufensystem	10
		Monitarisierung	10
	Hinzufügen des Shops		35
		Shop View	15
		Umsetzung von Items	20
	Einführung einer Währung mit Coins		30
		Design	10
		Balancing	20
	Regelmäßige Herausforderungen		15
		Aufstellen der Ziele	5
		Implementation	10
	Puffer		25
Stufe 4			150
	Implementierung von Features für zusätzliche Steigerung der Nutzerfreundlichkeit		40
		Filter für Projekte, Statistik und Achievements	15
		Sonstige	25
	Beseitigung der Bugs, welche in Alpha- und Beta-Phase aufgefallen sind		50
	Ausarbeitung		30
	Präsentation		30

3 Entwicklung

3.1 Projektadministration

3.1.1 Alphaversion

Die Entwicklung der Alphaversion der Plantivity-App zeigte einige Schwachstellen der Planung auf. Obwohl alle Must Have (bis auf die Illustration der Pflanzen sowie die *HallOfFameView*) und auch einige Nice to Have Arbeitspakete erfüllt sind, war die benötigte Anzahl der Stunden weitaus größer als ursprünglich angenommen, was eine Anpassung der Planung an dieser Stelle zur Folge hat. Die regelmäßigen Herausforderungen sind nun vollständig gestrichen, da sie zusätzlich als nicht relevant für die Game-Loop erkannt wurden und die Anzahl an geplanten Items ist auf das Wesentliche in Hinsicht auf UX reduziert.
Die Umsetzung erfolgt in einem täglichen Treffen mittels Pair-Programming. Die minimale Dauer des Treffens beträgt eine Stunde, es ist jedoch vorgesehen, das jeweilige Arbeitspaket für den Tag vollständig umzusetzen. Das erforderte jedoch, wie im obigen Absatz beschrieben, meist deutlich mehr Zeit. Durch das Pair-Programming ist zu begründen, dass die Deliverables direkt ins Projekt eingebunden werden, weshalb in diesem Kapitel nicht weiter auf diese eingegangen werden soll.
Des Weiteren gab es unerwartete Schwierigkeiten mit XCode, bzw. Swift, welche erhöhten Aufwand erfordert haben, da im Rahmen der Veranstaltung Serious Games kein Ansprechpartner für Fragen zur Swift-Entwicklung zur Verfügung steht. Dadurch sind einige bekannte Bugs in der Alphaversion bedingt, welche in Kapitel 3.2 weiter erläutert werden.

3.1.2 Betaversion

Die Problematiken der Zeitplanung der Alphaversion setzen sich für die Betaversion fort und konnten nicht vollständig kompensiert werden. Deshalb war weiterhin Zeit das größte Problem. Das hat die Auswirkung, dass auch in der Betaversion nicht alle der ursprünglich gesetzten Ziele erreicht wurden. Dieser Effekt ist gesteigert, da das Nachholen der in der Alphaversion ausgelassenen Arbeitspakete weitaus mehr Stunden in Anspruch genommen hat als erwartet.
Das Vorgehen in Pair-Programming wurde in der Betaversion ebenfalls auf die Probe gestellt, da das Zeichnen der Pflanzen weitaus zeitaufwändiger war, als erwartet. Somit wurden in der Betaversion nur zwei der geplanten Pflanzen umgesetzt, was dennoch die geplante Stundenzahl bereits überschritten hat. Da die Pflanzen höchste Priorität haben, ist auch das Fehlen der weiteren Illustrationen in der Hall of Fame und für Items zu begründen. Da dies die erste Erfahrung für Aufgaben aus dem Designbereich in Anwendungsentwicklung ist, hätte aus diesem Grund ein größerer Spielraum für die Umsetzung eingeplant werden sollen. Das wurde nicht getan, da sonst keine Zeit für weitere Features zur Verfügung gestanden hätte.

3.1.3 Tests

Eine Testgruppe von 5 Personen hat die App für eine Woche getestet. Die Tests sind dadurch eingeschränkt, dass durch XCode übertragene Applikationen nur begrenzt verwendet werden, woduch auch die informellen Treffen begründet sind, in welchen die Kritikpunkte genannt wurden. Diese wurden alle in die App eingearbeitet.

3.2 Beschreibung und Herangehensweise der Arbeitspakete der Alphaversion

In der Zeitplanung wurde die Entwicklung für die Alphaversion in einen Prototyp (Stufe 1) und die Alphaversion selbst (Stufe 2) unterteilt. Es wird in den folgenden Abschnitten immer genannt, in welcher der beiden Stufen die jeweilige Funktionalität umgesetzt wurde.

3.2.1 Datenstrukturen

Herangehensweise

In der ersten Stufe wurden die für die Funktionalität der App unerlässlichen Datenstrukturen für Projekte und Sessions in Core Data umgesetzt. Eine Schwierigkeit hierbei war, das Erstellen von Relationships für die gewünschte Projekthierarchie und der Zugehörigkeit von Sessions zu Projekten. Hierfür werden von XCode automatisiert Dateien erstellt, welche aber diese Relationships nicht wie gewünscht umgesetzt haben. Deshalb mussten diese manuell erstellt werden.
In der zweiten Stufe wurden des Weiteren Datenstrukturen für Pflanzen und Achievements umgesetzt, allerdings nicht in Core Data, da diese statische Objekte sind und daher nicht in der dynamischen Datenbank gespeichert werden müssen. Ebenfalls geplant war bereits die Datenstruktur für Items zu erstellen. Da dies aber nur als Nice-to-Have kategorisiert wurde und die Zeit sehr knapp bemessen war, ist das noch nicht erfolgt.

Beschreibung

Projekte und Sessions enthalten nun neben den unerlässlichen Attributen viele weitere Attribute, welche zur Speicherung von Daten für ausführliche Statistiken hilfreich sind. Dazu gehört etwa die Auswahl eines Icons zu Kategorisierung der Projekte. Dieses wird an einem späteren Punkt zur Verbesserung der visuellen Darstellung der Statistik verwendet.
Die Relationships wurden erfolgreich umgesetzt, sodass ein Projekt Unterprojekte und Sessions haben kann. Das erleichtert in Zukunft zusätzlich das Zusammenfügen von Projekten, da dabei nur das neue Zuordnen der zugehörigen Unterprojekte und Sessions eines Projekts zu einem anderen vorgenommen werden muss und anschließend das Projekt gelöscht werden kann. Da das allerdings nur als Nice-to-Have klassifiziert wurde, wurde dieses Feature aufgrund der Zeitknappheit ausgelassen.
Da das Design der Pflanzen noch fehlt, gibt es zu diesem Zeitpunkt nur Testpflanzen, allerdings ist die Datenstruktur nutzbar, sodass in der nächsten Stufe keine weitere Arbeit nötig ist, sobald die Designarbeit fertig ist.

3.2.2 Views

Herangehensweise

Für die Views erfolgt eine Unterteilung in beide Stufen. Für den Prototyp wurde mit der Umsetzung der Views für den Timer, die Projektansicht sowie zur Erstellung von Projekten die grundlegende Funktionalität sichergestellt. Dabei hat vor allem die *Timerview* Probleme bereitet. Die Anzeige der verbleibenden Zeit sollte möglichst präzise sein und deshalb auf die Sekunde genau angezeigt werden. Die View sekündlich zu aktualisieren ist allerdings in SwiftUI nicht direkt umsetzbar und hat dementsprechend die geplante Arbeitszeit überschritten. Des Weiteren ist es nach dem Prinzip der Least Clicks für die UX am sinnvollsten, den Picker für die ausgewählten Zeiteinheiten einer Sessions scrollable zu machen, da die Alternativen (etwa ein Drop-Down Menü oder manuelle Eingabe) wesentlich mehr Aufwand vom Nutzer erfordern. Allerdings ist im umgesetzten Picker in der aktuellen Version von SwiftUI ein Bug enthalten, sodass nur der am weitesten rechts liegende Picker verwendet werden kann. In der Alphaversion entspricht das der Eingabe für Sekunden, damit die Funktionalitäten leichter getestet werden können. Die Einstellung wird in der Finalen Abgabe nur auf die Minute genau sein. Das scheint dem Prinzip der Least Clicks widersprüchlich, jedoch fehlt die Option, kurze Sessions präzise einzustellen bei vielen bereits vorhandenen Tracking-Apps, was von einigen Nutzern als störend empfunden werden kann. Ein Kompromiss wird durch die Wahl des Pickers gefunden, welcher das Einstellen der Zeit einer Session durch Scrollen erlaubt.
Es ist anzumerken, dass die Umsetzung der *Projectsview* noch nicht fertig ist, da in der Alpha Version die Unterprojekte doppelt angezeigt werden. Einmal in der ausklappbaren Ansicht der Unterprojekte eines Projekts und zusätzlich als Überprojekt in der Projektliste. Das ist dadurch bedingt, dass noch kein Filter implementiert ist und obwohl die Datenstruktur bereits die nötigen Informationen zur Verfügung stellt, um Unterprojekte zu erkennen, ist eine Filterung für die korrekte Anzeige notwendig. Dieser wurde allerdings, da die Umsetzung der hierarchischen Anzeige bereits zeitaufwändig war und die Filterung ohnehin erst für eine spätere Stufe vorgesehen ist, noch nicht umgesetzt.

In der zweiten Stufe wurden weitere Views zur Ansicht von Statistiken und Achievements umgesetzt. Dies erfolgte ohne nennenswerte Komplikationen. Die Hall of Fame View, in welcher Pflanzen betrachtet werden können, ist noch nicht umgesetzt, da es ohne die Illustration der Pflanzen selbst keinen Sinn ergibt, für ein Zwischenergebnis Zeit aufzubringen.

Beschreibung

Es ist nun möglich, über eine Leiste am unteren Bildschirmrand die verschiedenen Views zu navigieren. Die *Projectsview* enthält erstellte Projekte und visualisiert den hierarchischen Zusammenhang dieser. Diese detailliertere Ansicht kann auch zu einer Gamification-fokussierten View mittels eines Buttons am oberen Bildschirmrand geändert werden. In dieser stehen die Pflanzen im Fokus und es ist nicht möglich, die Unterprojekte zu sehen. Der Sinn dieser View ist ein allgemeinerer Überblick über die Projekte.
Es ist ebenfalls möglich, in einer von der *Projectsview* aus erreichbaren weiteren View ein neues Projekt anzulegen und alle relevanten Einträge, wie etwa das übergeordnete Projekt, den Namen oder auch die Pflanze zu setzen. Diese View ist in einer leicht abgewandelten Form ebenfalls durch das Klicken auf ein Projekt in der Listenansicht der Projekte erreichbar, was das Bearbeiten und Löschen des Projekts erlaubt. So sind die weiteren CRUD-Operationen für Projekte umgesetzt. Der Timer ist ebenfalls funktionsfähig und visualisiert die verbleibende Zeit einer angefangenen Session, sodass der exakte Zeitwert nicht abgelesen werden muss.

3.2.3 Statistik

Herangehensweise

Die Statistik der App besteht aus einem Netzdiagramm, welche den Gesamtaufwand pro ausgewählter Kategorie anzeigt. Die Schwierigkeit dabei war, dass das Diagramm manuell in die View gezeichnet werden muss und abhängig von der Anzahl der ausgewählten Icons ist. Zusätzlich werden aber beim Abschließen von Sessions bereits mehr Daten erfasst, welche über einen Button in der Statistik View einsehbar sind. Dabei stand die visuelle Aufarbeitung nicht im Vordergrund, weshalb diese Erweiterung der Statistik zusätzlich umgesetzt werden konnte.

Beschreibung

Das Netzdiagramm zeigt die gearbeitete Zeit wie folgt an: Die Zeit aller für Projekte mit angezeigtem Icon absolvierten Sessions wird pro Projekt ermittelt und die höchste entspricht dabei 100%. Daraufhin wird die Zeit aller anderen Projekte damit ins Verhältnis gesetzt. Dadurch ergibt sich die Ausprägung der Anzeige pro Icon. Somit wird der Nutzer für ein ausgewogenes Verhältnis zwischen den verwendeten Kategorien motiviert, wodurch vermieden werden soll, dass manche Projekte vernachlässigt werden. Dieses Diagramm ist deshalb sehr Aussagekräftig, da der Nutzer aus einem Diagramm durch die verschiedene Auswahl an Icons viel Informationen erhalten kann. Da diese Art von Diagramm zusätzlich viel in Spielen zum Einsatz kommt und somit (im Gegensatz zu einem herkömmlichen Graph) ein Gefühl der Gamification vermittelt, fiel die Wahl für die erste visuelle Darstellung der Daten darauf.

3.2.4 Achievements

Herangehensweise

Die Achievements wurden einzeln implementiert und sind über das gesamte Projekt verteilt, um korrekt ausgelöst, bzw. überprüft zu werden. Als Beispiel gibt es Achievements, welche durch einen bestimmten Arbeitsaufwand (insgesamt oder bezüglich eines Projekts) verdient werden. Deren Bedingungen können in der *Timerview* nach dem Abschließen einer Session überprüft werden, um eventuell das Achievement herauszugeben. Andererseits kann auch ein Achievement durch das Abschließen oder Abbrechen eines Projekts ausgelöst werden, sodass zusätzlich in den entsprechenden Views hierfür Überprüfungen stattfinden. Diese Umsetzung erleichtert das Hinzufügen neuer Achievements, kann es allerdings unhandlich gestalten, ein Achievement zu entfernen, falls es sich doch als ungeeignet herausstellt.

Beschreibung

Die Achievements sind zu Beginn ausgegraut und es wird Nutzern kein Hinweis gegeben. Das Verfolgen spezifischer Achievements kann ein starker Motivator sein, falls dieses bekannt ist und steigert somit spezifische Aspekte der Produktivität. Allerdings werden einige Achievements sehr schnell erreicht und es ist beabsichtigt, dass so Nutzer allgemein motiviert bleiben, um mehr davon freizuschalten. Ein weiterer Nachteil von bekannten Meilensteinen ist, dass die Motivation schnell abnimmt, sobald alle erreicht sind. Somit können sehr schwer erreichbare, unbekannte Achievements für einen langen Zeitraum Motivation liefern. Falls ein Nutzer ein

Achievement erreicht, so ist dieses in der *Achievementsview* nicht weiter ausgegraut, sondern zeigt nun ein kleines Bild, den Titel sowie eine Beschreibung an, wofür das Achievement erhalten wurde.

3.3 Beschreibung und Herangehensweise der Arbeitspakete der Betaversion

3.3.1 Datenstrukturen

Herangehensweise

In der Betaversion haben sich die Datenstrukturen kaum verändert. Eine Datenstruktur wurde für Items angelegt. Dabei wurde wegen derselben Argumentation wie für Pflanzen auf Core Data verzichtet. Ebenfalls wurde die Datenstruktur von Projekten und Pflanzen angepasst, um die Verwendung der Pflanzen und Items zu erlauben.

Beschreibung

Die Datenstrukturen der Betaversion ermöglichen es, Items zu kaufen und zu verwenden. Die Zeit war jedoch nicht ausreichend, um die Funktionalität für alle Items umzusetzen. Das schließt das Arbeitspaket der Alphaversion ab, dafür jedoch das der Betaversion nicht vollständig.

3.3.2 Views

Herangehensweise

Um die Gamification zu erweitern, ist es sinnvoll, einen möglichst großen Fokus auf die Pflanzen zu legen. Dafür wurde die *HallOfFameView* umgesetzt. Diese fügt die bis zu diesem Punkt fehlende Shareability hinzu. Ebenfalls wurden die Projektansichten erweitert, um in der *ProjectView*, falls der Button auf die Pflanzenansicht gestellt ist, weiter die Gamification zu steigern. Es wurde darüber hinaus die Tabbar zum Navigieren der Views durch eine selbst erstellte ersetzt, da die durch SwiftUI gegebene Option nicht die Anforderungen hinsichtlich Design und Funktionalität erfüllt hat. Das hat eine Verbesserung der User-Experience zur Folge, da nun die *Timerview* noch zentraler steht und die Übersichtlichkeit verbessert wurde.
In der *StatisticView* gibt es nun eine weitere Statistik, welche eine Variante eines Balkendiagramms darstellt. Dabei wird wie auch beim Netzdiagram auf absolute Angaben verzichtet, sodass der Nutzer sich nicht auf subjektiv hohen Zahlenwerten ausruht, sondern weiter motiviert wird, seine Bestleistungen zu übertreffen. Zuletzt wurde die *Shopview* aktualisiert, sodass das Kaufen von Items möglich ist. Die Währung, mit welchen diese erworben werden können, sind die Coins, deren Betrag ebenfalls in der *Shopview* gesehen werden kann. Items, wie etwa zum Beleben von eingegangenen Pflanzen oder zum Einfrieren dieser, werden bewusst für Beträge verkauft, um Nutzer durch negative Motivation anzutreiben. Das hat den Effekt, dass wahrscheinlicher Ausnahmen vermieden werden und sich somit die Gewohnheit, regelmäßig an einem Projekt leichter bilden kann. Da die Pflanzen der App verschiedene Anspruchsstufen bieten, kann allerdings jeder Nutzer für sich entscheiden, wie sehr dieser Faktor einen Einfluss haben soll. Dadurch kann der Bedarf an Items minimiert werden, als Ausgleich allerdings auch die zu erwartende Belohnung pro Session dieser Projekte.

Beschreibung

Die *HallOfFameView* zeigt die Bestleistungen eines Nutzers. In der Betaverison fehlt ein Hintergrund, um die View ansehlich zu machen, jedoch werden die höchsten Stufen aller vom Nutzer verwendeten Pflanzenarten angezeigt. Die View repräsentiert somit einen objektiven, direkt ersichtlichen Status für andere Nutzer. Da die View geteilt werden kann, wird Shareability dadurch erhöht. So lassen sich mit wenig Aufwand die Leistungen von wetteifernden Nutzer vergleichen oder auch durch mehrere Posts auf Social-Media-Plattformen der Fortschritt von Projekten verfolgen.

3.3.3 Pflanzen

Herangehensweise

Die Umsetzung der Pflanzen gliedert sich in drei Bestandteile: Stufensystem, Belohnungssystem und Bilder der Pflanzen. Diese wurden in der Reihenfolge umgesetzt, wodurch auch die späte Erkenntnis, dass die Bilder länger dauern, als erwartet zu begründen ist. Das Stufensystem ist durch eine Anpassung der Datenstruktur und Berechnungen der aktuellen Stufe an geeigneten Stellen umgesetzt. Eine Schwierigkeit war dabei die konzeptionelle Umsetzung des Stufensystems. Die Betaversion enthält nun Pflanzen, welche sich in festgelegten Zeitintervallen aktualisieren und damit die aktuelle Stufe berechnen.
Das Belohnungssystem ist von der Stufe der Pflanze abhängig, weshalb die Umsetzung in dieser Reihenfolge sinnvoll zum Testen ist. Obwohl das Belohnungssystem noch nicht ausgewogen ist, erfordert dies sobald die Werte bekannt sind keinen weiteren Aufwand. Das ist der Fall, sobald die Tests abgeschlossen sind und einige Testnutzer[1] eine Rückmeldung geben.
Der letzte Abschnitt des Arbeitspakets der Pflanzen sind die Bilder. Die Umsetzung dieser erfolgte nicht immer in den Treffen für Pair-Programming, da die Vorteile des Pair-Programmings für die Erstellung der Bilder wegen der Geschwindigkeit der Umsetzung nicht vorhanden sind. Als Folge dessen wurden diese meist Teil der Freizeit erstellt, wodurch die exakte Arbeitszeit nicht bekannt ist.

Beschreibung

Der Nutzer kann nun seine mit Projekten verknüpfte Pflanzen durch das Fertigstellen von Sessions pflegen. Fällt die gearbeitete Zeit zu klein aus, steigt eine Pflanze ab, bis sie schließlich eingeht. Dabei geben höhere Stufen eine höhere Belohnung in Coins pro Zeiteinheit. Die Stufe der Pflanze wird durch ein sich änderndes Bild visualisiert. Diese Kombination hat das Ziel, möglichst viele Nutzer zu motivieren, die Stufen der Pflanzen zu erhöhen, da nicht nur die besseren Werte der Pflanzen einen Vorteil geben, sondern die größere Erscheinung zusätzlich ein Erfolgserlebnis geben kann.

[1] etwa fünf Personen aus dem Bekanntenkreis

3.4 Beschreibung und Herangehensweise der Arbeitspakete der finalen Version

3.4.1 User Experience Verbesserungen

Herangehensweise

Durch die Verwendung der Beta-Version und durch die Rückmeldungen der Testnutzer haben sich einige Vorschläge zur Optimierung der User-Experience gesammelt. Ebenfalls wurden einige verbleibende Punkte, wie etwa das Zusammenfügen von Projekten nachgeholt.

Beschreibung

Das in der Alphaversion angedeutete Feature, Projekte zusammenzufügen, wurde umgesetzt. Diese Funktionalität ist vorgesehen, da sich der Nutzer gerade beim Erstellen von Unterprojekten verschätzen kann und im Nachhinein feststellt, dass die Unterprojekte doch zusammengefasst werden sollten.
Eine dem Feedback entsprechend wichtige Neuerung ist ein Tutorial, um Nutzer mit den Funktionalitäten der App vertraut zu machen. Ein Nutzer sieht nach dem Öffnen der App die *Timerview*. falls er eine Session starten will, wird eine Meldung angezeigt, dass hierfür ein Projekt ausgewählt werden muss. Versucht der Nutzer nun eines auszuwählen, sieht er eine weitere Meldung, dass er noch keine Projekte hat. Um dies zu ändern, wird direkt ein Button angeboten, um das Prinzip der Least-Klicks einzuhalten. Da für das Erstellen von Projekten jedoch das der Pflanze entsprechende Item notwendig ist, wird der Nutzer auch darauf hingewiesen und weitergeleitet. Durch diese Form der Einführung lernt der Nutzer alle für die Benutzung relevanten Views, sowie deren Nutzen kennen. Beim Erstellen eines Projekts werden ebenfalls Nachrichten angezeigt, um Nutzer auf die Bestandteile eines Projekts hinzuweisen, falls diese nicht korrekt ausgefüllt sind.
Um in der *Project-, Statistic-* und *Achievementview* die Übersicht behalten zu können, kann ein Nutzer dort einen Filter, bzw. eine Namenssuche anwenden und erhält, falls aufgrund dessen die Anzeige leer ist, entsprechende Hinweise.
Zuletzt erweitert sich das Tutorial auf die Hall of Fame, in welcher der Nutzer ebenfalls eine knappe Anleitung erhält, wie diese zu füllen ist, sollte sie leer sein.

3.4.2 Hall of Fame Überarbeitung

Herangehensweise

Die Version der Hall of Fame View der Beta-Version hat durch fehlende individuelle Gestaltung die Shareability gesenkt. Da auch der geplante Hintergrund vorraussichtlich nicht in der nötigen Zeit fertiggestellt werden kann, um es in die finale Version zu schaffen, wurde das Konzept der Hall of Fame überarbeitet. Dabei wurde individualität in den Mittelpunkt gestellt und der Fokus eher auf einzelne Projekte gelegt, anstatt die allzeit Bestleistungen darzustellen. Da sich diese selten ändern, wird ein Nutzer nicht angeregt die Ansicht zu teilen, was die Shareability weiter senkt.
Dabei stellt das Speichern der Positionen der Pflanzen, sowie des Hintergrundbildes eine besondere Herausforderung dar, da dies nicht mithilfe der bisher verwendeten Methoden geschehen kann. Somit war hierfür das Einarbeiten in eine neue Thematik notwendig.

Beschreibung

Es ist Nutzern nun möglich, Projekte auszuwählen und die entsprechenden Pflanzen in der View zu platzieren. Mittels Drag-and-Drop können diese beliebig platziert werden, um so in Kombination mit einem vom Nutzer wählbaren Hintergrund die View größtmöglich zu individualisieren. Die View kann jederzeit bearbeitet werden und wird anschließend gespeichert.

3.4.3 Balancing der Pflanzen

Herangehensweise

Eine Herausforderung für die erfolgreiche Umsetzung der Belohnungen von Sessions und des Shops ist ein ausgeglichenes Währungssystem. Dieses basiert auf den verschiedenen Intervallen der Pflanzen und ist darauf ausgelegt, den Nutzer Pflanzen entsprechend der Größe des Projekts zu wählen. Eine Herausforderung war, dass Nutzer zu sehr versuchen wollen, das Spiel hinter der Tracking-App zu optimieren. Die Gamification soll der tatsächlichen Produktivität dienen und nicht ablenkend im Mittelpunkt stehen. Aus diesem Grund werden die genauen Daten dem Nutzer nicht gegeben und das Währungssystem so entworfen, dass Pflanzen auch nicht ideal verwendet werden können, ohne die User Experience zu beeinträchtigen.

Beschreibung

Für die Version, welche einige Testnutzer erhalten haben, war bei einem durchschnittlichen Arbeitsaufwand von einer Stunde pro Tag innerhalb des nächst größeren Update Intervalls die für die Pflanze ausgegebenen Coins zurück. Dies motiviert dazu für Projekte von etwa einer Woche die Orchidee zu nehmen, bei einer Dauer von etwa einem Monat die Sonnenblume. Für Langzeitprojekte bis zu einem Jahr eignet sich der Kaktus und für Projekte darüber hinaus der Banianbaum. Dieses Konzept soll dabei helfen, dass die Hall of Fame nicht monoton aussieht, was der Fall ist, wenn es eine objektiv beste Pflanze gibt. Einen Sonderfall stellt jedoch die Sonnenblume dar, da diese kostenlos ist, damit sich ein Nutzer nicht versehentlich in ein Deadlock ohne Pflanzen und ohne Geld schließen kann. Als Konsequenz hat es sich ergeben, dass die Belohnungen niedriger ausfallen müssen. Nach dem Feedback der Nutzer wurden die Preise und Belohnungen der Pflanzen leicht angepasst.

4 Zusammenfassung und Ausblick

4.1 Ergebnis

4.1.1 Arbeitspakete

Alle *Must-Have* Arbeitspakete wurden umgesetzt. Ein *Nice-To-Have* Arbeitspaket, tägliche Herausforderungen, welches im Lauf der Entwicklung als nicht relevant für die Gamification und auch Funktionalität der App angesehen wurde, wurde, um die Überschreitung der geplanten Stundenzahlen anderer Arbeitspakete zu kompensieren und auf den zusätzlichen Rat des Betreuers aus dem Projektplan entfernt. Alle weiteren *Nice-To-Have* Arbeitspakete wurden ebenfalls implementiert.

4.1.2 Zeitplanung

Die geplante Stundenzahl wurde für nahezu jedes Arbeitspaket überschritten, um die gesetzten Fristen des Gantt-Diagrams einzuhalten. Zu nennen ist hierbei in Stufe 1 die *TimerView* und Umsetzung der Datenstrukturen und Relationen von, bzw. zwischen Projekten und Sessions. In Stufe 2 wurde das Erstellen der Assets für die Pflanzen, Items und Achievement-Icons nicht umgesetzt, da die verbleibenden Views und das Berechnen und Visualisieren der Statistiken die Stundenanzahl überschritten haben. Das Entfernen der sich in Stufe 3 befindenden täglichen Herausforderungen hat das Nachholen der letzten Aspekte aus der vorherigen Stufe ermöglicht. Dabei wurde allerdings die geplante Anzahl an gezeichneten Pflanzen nicht erreicht, da das Erstellen der Assets ohne Vorerfahrung geschah. Da in einem Gespräch mit dem Betreuer Kritik an der Hall of Fame geübt wurde, nahm die Überarbeitung dieser den Großteil von Stufe 4 ein, weshalb diese ebenfalls die geplante Bearbeitungsdauer überschritten hat.

4.2 Highlights

Highlights der App hinsichtlich des Lerneffekts für iOS-Development sind unter anderem die selbst erstellten Animationen, wie die des Timers und der Tabbar oder auch selbst erstellte UI Elemente, wie Slider, Themes, Popups, Statistiken und ebenfalls die Tabbar. Das Erstellen, bzw. Einfügen der Assets hat eine besondere Herausforderung dargestellt, da noch keine Erfahrung bestand und online wenig Hilfe zu finden war. Dennoch handelt es sich dabei um eine wichtige Fähigkeit in der App-Entwicklung.

4.3 Offene Punkte und Erweiterungen

Dass die vorgesehenen Arbeitspakete die geplante Zeitdauer nahezu immer überschritten haben, war in den ersten drei Stufen kein Problem, weil sich das Projekt über den Rahmen des Moduls als Hobby entwickelt hat. Aus diesem Grund war auch die Ambition vorhanden, die App über die Bewertungskriterien hinaus zu verbessern. deshalb ist für die Semesterferien weitere Arbeit, und damit nach der Veröffentlichung für das Modul zeitnah ein Update geplant.

Dabei sind die Illustrationen der verbleibenden Pflanzen vorgesehen, da ohne diese der Gamification-Aspekt nicht sein volles Potenzial hat. Ebenfalls sind eigene Bilder für die Pflanzen im Shop vorgesehen. Neben diesen wurde auch die Umsetzung weiterer Items durch das langsame Erstellen der Assets limitiert. Demnach werden geplante Items wie etwa ein Bonus für Sessions im Nachhinein hinzugefügt. Das Hinzufügen eines Standard-Hintergrundbildes in der Hall of Fame kann ebenfalls zu leicht mit der App identifizierbaren Bildern führen. Das würde den Nutzen der Shareability Features steigern, da die App erkennbar wird und Nutzer eher ein Bild teilen, wenn sie nicht selbst ein Bild einfügen müssen. Da nach dem Gamification-Aspekt die Statistiken einen Großteil der Motivation der App erzeugen, ist auch eine Erweiterung der Visualisierung und Optionen in der *Statistikview* sinnvoll. Das hilft Nutzern, einen direkteren und hilfreicheren Überblick über die erbrachten Leistungen zu erhalten. Da sich das Framework *Swift Charts* gerade noch in der Beta befindet, werden die Statistiken überarbeitet und erweitert, sobald es veröffentlicht wird. Des Weiteren ist auch eine lange Liste an kleinen Verbesserungen geplant. Für Nutzer, welche weniger Interesse an den Statistiken haben, soll beispielsweise das Erstellen mehrerer Themes, welche durch Coins freischaltbar sind, die Ambition steigern. Hierbei werden auch von Nutzern frei erstellbare Themes in Betracht gezogen.