Projektdokumentation

Dominic Jackson, Leszek Gurczynski

Fachinformatiker für Anwendungsentwicklung

m-lease GmbH, DiCentral GmbH

Android Projekt: Spiel „Mensch Ärgere dich nicht“

22.01.20

Inhalt

[3. Projektauftrag 3](#_Toc30686249)

[3.1 Projektumfeld 3](#_Toc30686250)

[3.2 Projektziel 3](#_Toc30686251)

[3.3 Projektabgrenzung 3](#_Toc30686252)

[4. Projektplanung 4](#_Toc30686253)

[4.1 Allgemeine Beschreibung der Funktionalität der App 4](#_Toc30686254)

[4.2 Beschreibung des Ist-Zustandes 4](#_Toc30686255)

[4.3 Beschreibung des Soll-Konzepts 4](#_Toc30686256)

[4.4 Vorgehensmodell 4](#_Toc30686257)

[4.5 Zeitplanung 4](#_Toc30686258)

[5. Projektdurchführung 5](#_Toc30686259)

[5.1 Implementierung 5](#_Toc30686260)

[5.2 Qualitätssicherung 5](#_Toc30686261)

[6. Projektabschluss 6](#_Toc30686262)

[6.1 Soll-Ist-Vergleich 6](#_Toc30686263)

[6.2 Kostenanalyse 6](#_Toc30686264)

[6.3 Fazit 6](#_Toc30686265)

[6.4 Ausblick 6](#_Toc30686266)

[7. Anhang 7](#_Toc30686267)

[7.1 Quellen 7](#_Toc30686268)

[7.2 Glossar 7](#_Toc30686269)

[7.3 Anhang (UML-Diagramme, Screenshots, Quellcode) 7](#_Toc30686270)

# 3. Projektauftrag

* Projektname: Android Spiel „Mensch ärgere dich nicht“
* Ausgangsituation (vgl. 3.1)
* Projektziele- und Nicht-Ziele (vgl. 3.2/ 3.3)
* Projektnutzen (vgl. 3.2)
* Start- und Endtermin
* Projektteam: Dominic Jackson, Leszek Gurczynski

## 3.1 Projektumfeld

Der Lehrer an dem Staatlichen Beruflichen Schulzentrum Freising Herr Albrecht Pfleiderer gibt den Arbeitsauftrag an Dominic Jackson (DiCentral GmbH), Leszek Gurczynski (m-lease GmbH) im Rahmen der Lehrveranstaltung Anwendungsentwicklung und Programmierung eine App in der Android Studio Umgebung zu entwickeln.

Die Anwendung soll die Login-Daten von Anwendern abspeichern, und die Kommunikation zwischen mehreren Anwendern ermöglichen. Aus diesem Grund wurde die Entwicklungs-Plattform Firebase verwendet.

Die Anwendung wird in Android Studio DIE entwickelt und getestet. Zur Testumgebung gehört außer den AVDs das im Unterricht zur Verfügung gestellte Endgerät mit Android OS Motorola One Macro.

## 3.2 Projektziel

Die zu erreichende Funktionalität ist eine Spiel-Anwendung, die nicht nur der Singleplayer Modus anbietet, sondern auch die Interaktion mit mehreren Anwendern unterstützt.

## 3.3 Projektabgrenzung

Das Projekt soll am 18.10.19 starten. Die Abgabefrist ist am 24.01.20. Im Rahmen des Projekts wird das Spiel „Mensch ärgere dich nicht“ entwickelt. Zum Teil des Projekts gehört es nicht, eine Datenbank zu entwickeln, um die Login-Daten von Anwendern zu speichern. Der Lehrer Herr Albrecht Pfleiderer wird regelmäßig über erreichte Milestones benachrichtigt, und es ist möglich für ihn, das Projekt im Git Repository zu verfolgen.

# 4. Projektplanung

## 4.1 Allgemeine Beschreibung der Funktionalität der App

Die Anwendung ermöglicht es den Benutzern, das bekannte Brettspiel „Mensch ärgere dich nicht“ auf einem Endgerät mit Android OS zu spielen. Es sind drei Spielmodi verfügbar: Single-, Multiplayer- (Offline), und Multiplayer (Online). Die Anwender haben Möglichkeit entweder einen Account mit einer E-Mail-Adresse und Passwort anzulegen oder als Gast zu spielen. Bei der Wahl als Gast werden keine Login-Daten von den Anwendern erhoben. Nach diesem Vorgang erstellt ein Spieler ein neues Spielbrett oder tritt einem bei.

Die Spielregeln lauten folgedermaßen:

Wer eine Sechs würfelt, kann eine eigene Spielfigur aus der Startposition heraus auf sein Startfeld stellen; danach darf er erneut würfeln und mit der Figur entsprechend viele Felder vorrücken. Das Startfeld muss so bald wie möglich wieder freigemacht werden. Hat er jedoch keine Figur mehr in der Startposition, so steht es ihm frei, die erwürfelten sechs Felder mit einer Figur seiner Wahl vorzurücken. Auch dann darf er erneut würfeln und einen weiteren Zug machen.

Kommt beim Umlauf eine Spielfigur auf ein Feld, das bereits von einer gegnerischen Spielfigur besetzt ist, gilt die gegnerische Figur als geschlagen und muss zurück auf ihre Startposition.

Eigene Figuren können nicht geschlagen werden: Ist das Zielfeld bereits mit einer eigenen Figur besetzt, darf der Zug weder ausgeführt noch neu gewürfelt werden, sondern es ist der nächste Spieler an der Reihe. Das war nicht immer so, die Spielregel zur Ausgabe von 1960 sagt zum Einsetzen einer Figur auf den Anfangskreis „A“ nach dem Würfeln einer 6: „[…] befindet sich dort schon eine Figur seiner Farbe, so muss er mit derselben 6 Kreise weiterziehen, selbst wenn er auch eine eigene Figur dadurch hinausschlägt.“

Hat ein Spieler mehrere Spielfiguren im Umlauf, kann er frei entscheiden, mit welcher er ziehen möchte. Ein Würfelwurf darf allerdings nicht auf mehrere Figuren aufgeteilt werden.

Wenn man mit seiner letzten Figur vor seinem Haus steht und eine Sechs würfelt, kann man nicht fahren und somit gilt der Zug als beendet.

## 4.2 Beschreibung des Ist-Zustandes

Zu Beginn des Projekts ist eine im Unterricht entwickelte Beispiel App, Android Studio Dokumentation und diverse Anleitungen bezüglich des Funktionalitätsausbaus der App im Form PDF-Dateien vorhanden. Eine tiefe Einarbeitung in das Thema Entwicklung von Anwendung in der Android Studio Umgebung ist erforderlich.

## 4.3 Beschreibung des Soll-Konzepts

Im Rahmen des Projektes soll die Spiel-App „Mensch ärgere dich nicht“ entwickelt werden. Die Hauptfunktion der App ist es, dem Benutzer es ermöglichen, das bekannte Brettspiel auf Endgerät zu spielen. Die Funktionalität der App wurde in Java-Klassen zerlegt, und der Lösungsweg in jeweiligen Klassen abgebildet. Es wird streng wie möglich nach dem DRY-Prinzip entwickelt, und jede Klasse soll genau eine definierte Aufgabe erfüllen.

Um den hohen Zeitaufwand bei Entwicklung eines eigenen Datenbankserver zu sparen soll die Entwicklungsplattform Firebase genutzt werden.

## 4.4 Vorgehensmodell

Im Verlauf des Projekts wird nach dem inkrementellen Vorgehensmodell vergangen. Jede Funktionalität wird schrittweise implementiert und getestet. Das Festlegen von Mile Stones wurde durchgeführt.

## 4.5 Zeitplanung

|  |  |
| --- | --- |
| Phase | Zeit |
| Analysephase | 20h |
| Entwurfsphase | 45h |
| Implementierungsphase | 78 h |
| Testphase / Qualitätssicherung | 30 h |
| Gesamt | 173h |

# 5. Projektdurchführung

## 5.1 Implementierung

Erstellen von User mit 2 Parameter: -E-Mail und Password. Die User-Daten werden hiermit in der Firebase Datenbank gespeichert.

Die hier verwendete Methode stammt von der Klasse Firebaseauth.

**firebaseAuth**.createUserWithEmailAndPassword(email, password)  
.addOnCompleteListener(**this**, **new** OnCompleteListener<AuthResult>() {  
 @Override  
 **public void** onComplete(@NonNull Task<AuthResult> task) {  
 **if** (task.isSuccessful())  
 {  
 Toast.*makeText*(CreatUser.**this**, **"Erforlgreich registriert"**, Toast.***LENGTH\_SHORT***).show();  
 } **else** {  
 Toast.*makeText*(CreatUser.**this**, **"Registrierung fehlgeschlagen"**, Toast.***LENGTH\_SHORT***).show();  
 }  
 }  
 }  
);

**private void** createSpielfeld() {  
 setSpielfeld();  
 createSpiel();  
 addSpieler();  
 setFigur();  
 gameLoop();  
  
}

Die Klasse Spiel erzeugt eine Klasse CreateSpiel diese beinhaltet die Klasse Spielfeld, diese ist für die Positionen und setzen Parameter der 72 Spielfelder zuständig.

Außerdem erzeugt die Klasse Spiel noch ein Array neue Spieler in dem die einzelnen Spieler und der Status deren Figuren gespeichert und angepasst wird, die Positionierung und Darstellung der Felder ist in der Kasse Spiel realisiert.

Diese erzeugt einen neuen Thread der für die Steuerung des Spieles zuständig ist.

**private void** createSpiel() {  
  
*main* = **new** CreateSpiel(**spieler**, **bildschirmBreiteMitte**, **bildschirmHoeheMitte**, **rSPielfeldHoeheBreite**);  
  
 **for**(**int** i = 0; i < 72; i++) {  
 **int** x = *main*.getFelder(i).getX();  
 **int** y = *main*.getFelder(i).getY();  
  
 **feld** = **new** ImageView(**this**);  
  
 **switch** (*main*.getFeldbesitzer(i)) {  
 **case** 0 : **feld**.setImageResource(R.drawable.***feld***);  
 **break**;  
 **case** 1: **feld**.setImageResource(R.drawable.***blau***);  
 **break**;  
 **case** 2: **feld**.setImageResource(R.drawable.***gelb***);  
 **break**;  
 **case** 3: **feld**.setImageResource(R.drawable.***gruen***);  
 **break**;  
 **case** 4: **feld**.setImageResource(R.drawable.***orange***);  
 **break**;  
 }

FrameLayout.LayoutParams feldParams = **new** FrameLayout.LayoutParams((**int**) (*SpielfeldHoeheBreite* / 16.5), (**int**) (*SpielfeldHoeheBreite*/16.5));  
  
 feldParams.**leftMargin** = (x);  
 feldParams.**bottomMargin** = (y) ;  
 feldParams.**gravity** = Gravity.***BOTTOM*** + Gravity.***LEFT***;  
  
 **spielbereich**.addView(**feld**, feldParams);  
 }  
}

## 5.2 Qualitätssicherung

Die Tests wurden direkt nach der Implementierung von jeweiligen Methoden bzw. nach der Umsetzung der Logik durchgeführt. Die hier verwendete Test-Methode heißt Schreibtisch Test.

Zudem wurden die beiden Entwickler durch die umfangreiche Debugging Mechanismen von Android Studio und aussagekräftige Logs, die zur Laufzeit generiert wurden, unterstützt.

# 6. Projektabschluss

## 6.1 Soll-Ist-Vergleich

Das Projekt konnte wie geplant in ~170 Stunden abgeschlossen werden, es konnten allerdings mehre Zeit Abweichungen festgestellt werden. Der zeitliche Aufwand der Implementierung sowie der Konzeption wurde unterschätzt, denn notwendige Einarbeitung in die Android Studio Umgebung erforderlich war.

|  |  |  |  |
| --- | --- | --- | --- |
| Phase | Zeit geplant | Zeit realistisch |  |
| Analysephase | 20h | 10h |  |
| Entwurfsphase | 15h | 7h |  |
| Implementierungsphase | 78 h | 92h |  |
| Testphase / Qualitätssicherung | 30 h | 15h |  |
| Gesamt | 173h | 124h |  |

## 6.2 Kostenanalyse

Die Tabelle stellt den Dienstleistungsaufwand dar. Es konnten die Gesamtkosten in Höhe von 3224,00€ berechnet werden.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Vorgang | Mitarbeiter | Zeit | Kostenpauschale pro Stunde | Gesamtkosten |
| Entwicklung | Auszubildender x2 | 124 | 13€ | 3224,00€ |
|  |  |  |  |  |

## 6.3 Fazit

Nach Abschluss des Projekts ziehen die Auszubildende ein persönliches Fazit. Das Gelernte wird reflektiert und ein Ausblick auf die zukünftige Arbeit in IT-Projekten, bezogen auf die Auswirkungen des Projekts, wird gegeben.

## 

# 7. Anhang

## 7.1 Quellen

https://de.wikipedia.org/wiki/Mensch\_%C3%A4rgere\_Dich\_nicht

## 7.2 Glossar

* AVD – Android Virtual Device

## 7.3 Anhang (UML-Diagramme, Screenshots, Quellcode)