**Projektarbeit Informatik Q11: Roulettespiel**

Teammitglieder: Zollitsch Christoph, Haberl Simon, Sauer Simon, Erben Emanuel

Untergliederung in zwei Gruppen:- SQL-Datenbankanbindung: Simon S. & Simon H.

-Graphische Oberfläche: Christoph und Emanuel

**Wasserfallmodel:**

**1.Analyse:**

Ziel der Projektarbeit ist es ein Roulettespiel mit einer Anmeldeoberfläche zu entwickeln.

Ferner sollen Benutzerdaten (u.a. Benutzername und Passwort) und das Verlaufsprotokoll des Spiels in einer Datenbank verwaltet werden.

Geplanter Spielablauf:

Nach erfolgreicher Anmeldung bzw. Registrierung auf einer Anmeldeoberfläche, erscheint die Spieloberfläche: Auf der Spieloberfläche erscheinen das Roulettefeld, die Auswahl des Einsatzes in Form von Chitons, der Roulettedreher und personenbezogene Daten wie beispielsweise aktuelles Guthaben und Gewinn. Durch betätigen des Abmeldebuttons wird der Benutzer auf die Anmeldeoberfläche zurück geleitet.

Zustandsübergangsdiagramm:

**2.Entwurf:**

Klassendiagramm:

**3.Implementierung:**

Graphische Oberfläche:

Zuerst haben wir einfache Copyright freie Bilder für die Anmeldeoberfläche und das Spielpanel gesucht. Hierbei haben wir zwei gute Bilder gefunden, die zwar eine gute Qualität hatten, jedoch meist nur eine Auflösung von 960 auf460 Pixel hatten. Wenn man jetzt die Größe verändert hätte, wären Weiße Ränder entstanden, die wir jedoch verhindern wollten. Deshalb haben wir in unser bereits erstellter MVC- Struktur, in der Klasse Login die Zeile „setResizable(false);“ hinzugefügt, wodurch die größe unseres JPanel´s nicht mehr verändert werden kann. Danach wollten wir anfangen auf der Anmeldeoberfläche eine Eingabefläche für Benutzername und Passwort hinzuzufügen. Hierbei wollten mussten wir uns jedoch ersteinmal für ein Layoutmanager entscheiden. Jedoch haben wir kein Layout gefunden, dass mit unseren Anforderungen zusammenpasste, haben wir uns dafür entschieden, kein Layoutmanager zu verwenden, und alle Bestandteile unserer Oberfläche einzeln an genauen Positionen zu platzieren. Dies war auch kein Problem, da ja die größe des Frames nicht verändert werden kann, und somit auch die Buttons und andere Elemente auf dem Frame nicht verrutschen können.

SQL-Datenbankanbindung:

Am Anfang wollten wir Access als Datenbank und den vorgegebenen Programmtext zur Datenbankverbindung nutzen. Allerdings stießen wir dabei auf einige Probleme: Der Access-Treiber konnte nicht geladen werden, weshalb wir versuchten im Internet Lösungen zu finden. Wir konnten aber keinen geeigneten Treiber laden bzw. finden. Deshalb erstellten wir eine neue Testklasse mit einer Methode, in der wir alle Treiber auflisteten, die uns zur Verfügung stehen. Daraufhin entschieden wir uns anhand des Treibers für eine neue Datenbank: SQLite, die wir mithilfe einer portablen Variante nutzten, damit wir sowohl in der Schule als auch zu Hause effektiv am Projekt arbeiten konnten.