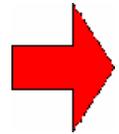


Projektaufgabe

Multimediateprogrammierung

Inhalt



1. Vorstellung der Aufgabenstellung
2. Organisatorisches
3. Einteilung der Teams
4. Vorstellung benötigter Werkzeuge
 1. Versionsmanagement mit Subversion
 2. Automatisierte Benutzereingaben mit Autolt
 3. Electro-Server
5. Einstiegsaufgabe

Aufgabenstellung

- Erstellung eines Projektes mit Flash/ActionScript
- Spiel: Rennen von Fahrzeugen, z. B. Autorennen
- Möglichst realistisches Fahrverhalten
- Verschiedene Fahrzeuge, Strecken und Spielmodi
- Mehrspieler-Modus
- Aufgeführte Anforderungen zunächst nur „Wunschliste“ und Anregung, konkrete Anforderungen werden mit Kunden festgelegt!

Strecke

- Verhalten der Fahrzeuge je nach Bodenbelag (z. B. Abbremsen beim Verlassen der Strecke usw.)
- Beschädigung des Fahrzeuges beim Aufprall auf Hindernisse oder andere Fahrzeuge.
- Auftanken oder Reparieren („Boxenstopp“) der Fahrzeuge an bestimmten Stellen der Strecke möglich
- Verschiedene Strecken
- Erweiterung der Anwendung um weitere Strecken möglichst einfach möglich

Fahrzeuge

- Angemessen realistisches Bewegungsverhalten
- Verschiedene Fahrzeuge mit unterschiedlichem Fahrverhalten
- Sound für Fahrgeräusche
- Erweiterung der Anwendung um weitere Fahrzeuge möglichst einfach möglich

Anzeige

- Möglichst komfortable Ansicht des Fahrzeugs (eventuell auch mehrere Ansichten)
- Fahrzeugzustand + „Cockpit“
- Gefahrene Runden
- Zeiten (Rundenzeit, Gesamtzeit usw.)

Einspieler-Modus

- Gegnerische Fahrzeuge
- Verschiedene Level und Modi (z. B. Trainingsmodus und Wettbewerbsmodus)
- Bestenlisten

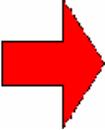
Mehrspieler-Modus

- Einloggen auf dem Server
- Sichtbarkeit anderer eingeloggter Spieler
- Austausch von Nachrichten unter den Spielern (Chatten)
- Eröffnen neuer Spiele, an denen sich die anderen Spieler beteiligen können
- Spielablauf:
 - Mindestanforderung: mehrere Spieler fahren gleichzeitig auf separaten Strecken und es werden nur die Zwischenzeiten übertragen
 - Wunsch-Anforderung: Möglichst mehrere Spieler fahren in Echtzeit gleichzeitig auf der gleichen Strecke und können sich gegenseitig sehen, miteinander kollidieren.

Weitere Anwendungs-Bestandteile

- Animiertes Intro
- Hilfe und Info über Anwendung und Autoren
- Sound

Inhalt

1. Vorstellung der Aufgabenstellung
-  2. Organisatorisches
3. Einteilung der Teams
4. Vorstellung benötigter Werkzeuge
 1. Versionsmanagement mit Subversion
 2. Automatisierte Benutzereingaben mit Autolt
 3. Electro-Server
5. Einstiegsaufgabe

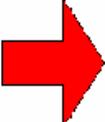
Voraussetzungen

- Team mit 6 Teilnehmern
- Vorgehen nach Extreme-Programming-Ansatz
- Team bestimmt Projektleiter
=> ist Ansprechpartner nach Außen
- Pro Team 1 Betreuer
 - regelmäßige Treffen, ca. 1x pro Woche
 - übernimmt Rolle des Kunden
- Bei jedem Treffen wird mittels Planning Game festgelegt, welche Anforderung als nächstes realisiert werden soll und bis wann
- Endergebnis der Projektaufgabe muss nicht produktionsreif sein

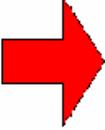
Team-Aufgaben

- Zeitabstimmung
- Aufteilung der Aufgaben (aber XP: jeder macht jedes)
- Keine 40-Stunden-Woche sondern 40-Stunden-Projekt (als grobe Richtlinie)
- Notation der einzelnen Schritte:
 - Iterationen und User Storys
 - Benötigte Stundenzahl
 - Probleme...

Inhalt

1. Vorstellung der Aufgabenstellung
2. Organisatorisches
-  3. Einteilung der Teams
4. Vorstellung benötigter Werkzeuge
 1. Versionsmanagement mit Subversion
 2. Automatisierte Benutzereingaben mit Autolt
 3. Electro-Server
5. Einstiegsaufgabe

Inhalt

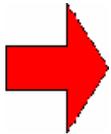
1. Vorstellung der Aufgabenstellung
2. Organisatorisches
3. Einteilung der Teams
-  4. Vorstellung benötigter Werkzeuge
 1. Versionsmanagement mit Subversion
 2. Automatisierte Benutzereingaben mit Autolt
 3. Electro-Server
5. Einstiegsaufgabe

Werkzeuge

- Versionsmanagement mit Subversion:
Siehe separate Folien...
- AutoIt (<http://www.autoitscript.com/autoit3/>)
Siehe Vorführung...

Inhalt

1. Vorstellung der Aufgabenstellung
2. Organisatorisches
3. Einteilung der Teams
4. Vorstellung benötigter Werkzeuge
 1. Versionsmanagement mit Subversion
 2. Automatisierte Benutzereingaben mit Autolt
 3. Electro-Server
5. Einstiegsaufgabe



Electro-Server: Grundaufbau

- <http://www.electrotank.com/electroserver/>
- Auf dem Server:
 - Java-Anwendung, plattformunabhängig
 - Einmalig parametrisieren (mit IP-Adresse und Port)
 - Starten => wartet auf Nachrichten von Clients
 - Sendet empfangene Nachrichten an alle Clients weiter
- Client-Anwendung:
 - ActionScript-Klasse *ElectroServer* in Anwendung einbinden
 - Methoden der Klasse aufrufen, diese erledigt die Kommunikation mit dem Server
 - Wird aufgerufen, falls Nachricht vom Server eintrifft

ElectroServer-Klasse: Anwendung

- Prinzip:
 - Nachrichten an Server senden durch einfachen Methodenaufruf
 - Nachrichten von Server erhalten: Methode zur Ereignisbehandlung bereitstellen (überschreiben)
- Instanz von ElectroServer-Klasse holen mit *getInstance()* (Entwurfsmuster Singleton)

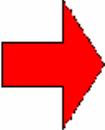
Beispiel: Verbinden mit Server

- Methode *onConnection(success:Boolean, error:String)* überschreiben:
 - Falls *success = true* dann weiter im Programmablauf
 - Ansonsten: Verbindung konnte nicht hergestellt werden, Fehlermeldung *error* ausgeben
- Parameter setzen mit *setIP(ip)* und *setPort(p)*
- *connect()* aufrufen
- Server meldet sich zurück, indem obige Methode *onConnection()* aufgerufen wird

Weitere Schritte

- Weitere Schritte nach gleichem Prinzip:
 - Nutzer einloggen mit Nutzernamen
 - Andere eingeloggte Nutzer abfragen
 - Vorhandene Chat-Rooms abfragen
 - Chat-Room beitreten oder neuen anlegen
 - Chat-Nachricht senden
 - Game-Room beitreten oder neuen anlegen
 - (Spiele-)Daten senden
- Game-Room: Spezialfall von Chat-Room, in dem nicht Chat-Nachrichten ausgetauscht werden, sondern beliebige (Spiele-)Daten

Inhalt

1. Vorstellung der Aufgabenstellung
2. Organisatorisches
3. Einteilung der Teams
4. Vorstellung benötigter Werkzeuge
 1. Versionsmanagement mit Subversion
 2. Automatisierte Benutzereingaben mit Autolt
 3. Electro-Server
-  5. Einstiegsaufgabe

Start-Aufgabe

- Abstimmung des Teams (Kommunikationswege, zeitliche Abstimmung)
- Festlegen des Projektleiters
- Aufgabenstellung und Umsetzung der XP-Prinzipien durchsprechen
- Anregungen: Buch *Games Most Wanted*
- Erstellen eines Grundgerüsts für die Anwendung:
 - Aufteilung des Hauptbildschirms
 - Festlegung auf Fahrzeuge
 - Grundlegende Implementierung eines Fahrzeugs
 - Grundlegende Implementierung einer Strecke