

**UE Software Engineering 050052
WS 2016/17**

Gruppe:
LV-Leiter:

Projektname: E-Honk

Projektteam:

| Nachname | Vorname | Matrikelnummer | E-Mail-Adresse |
|---------------|---------|----------------|----------------------------|
| Aschenbrenner | Florian | 0000805 | a0000805@unet.univie.ac.at |
| Baidiuk | Yevhen | 1368277 | a1368277@unet.univie.ac.at |
| Kuzmenko | Oksana | 1369160 | a1369160@unet.univie.ac.at |
| Räth | Martin | 1263997 | a1263997@unet.univie.ac.at |

CEWebS-Teamseite:

https://cewebs.cs.univie.ac.at/SWE/ws16/index.php?m=D&t=uebung&c=show&CEWebS_c=g050052-3t2

Datum: 23.11.2016

1. Funktionale Anforderungen

- *Brainstorming*
- *Befragung von Endbenutzern (Interviews)*
- *Analogien (Erfahrungen aus gleichartigen Systemen)*
- *Erfahrungen aus eigener Hand*

1.1. Funktionalität

Es ist geplant eine leicht verständliche Webplattform zu erstellen, welche es allen Usern ermöglicht Dinge zu verkaufen oder zu kaufen. Dies benötigt eine Registrierung aller Benutzer, welche mit einem Formular durchgeführt wird. Es werden die wichtigsten Daten der User gespeichert, Vorname, Nachname, E-Mail-Adresse, (evtl. Geschlecht, Telefonnummer), und das Passwort.

Beim Login wird der Username und das Passwort abgefragt. User sollen auch die Möglichkeit bekommen ihr Nutzerprofil nachträglich zu ändern, somit ihren bestehenden Eintrag in der Datenbank zu löschen, oder einzelne Daten zu ändern. Es ist ebenfalls geplant eine Suchfunktion für Artikel zur Verfügung zu stellen.

Käufer sollen die Möglichkeit haben Gebote auf verschiedene Artikel abzugeben. Weiters soll eine Watchlist für bereits vorliegende Gebote zur Verfügung gestellt werden, bzw. ist geplant diese zu erweitern auch für Auktionen/Artikel, auf welche der User nicht geboten hat.

Verkäufer sollen die Möglichkeit haben Artikel einzustellen. Das inkludiert für jeden Artikel eine Beschreibung, einen Startpreis, einen Sofortkaufpreis, einen aktuellen Preis, eine auszuwählende Kategorie und einen Namen. Es soll möglich sein den Preis zu ändern sofern noch keine Gebote vorliegen und die Beschreibung und den Namen anzupassen. Artikel sollen auch als verkauft markiert werden können.

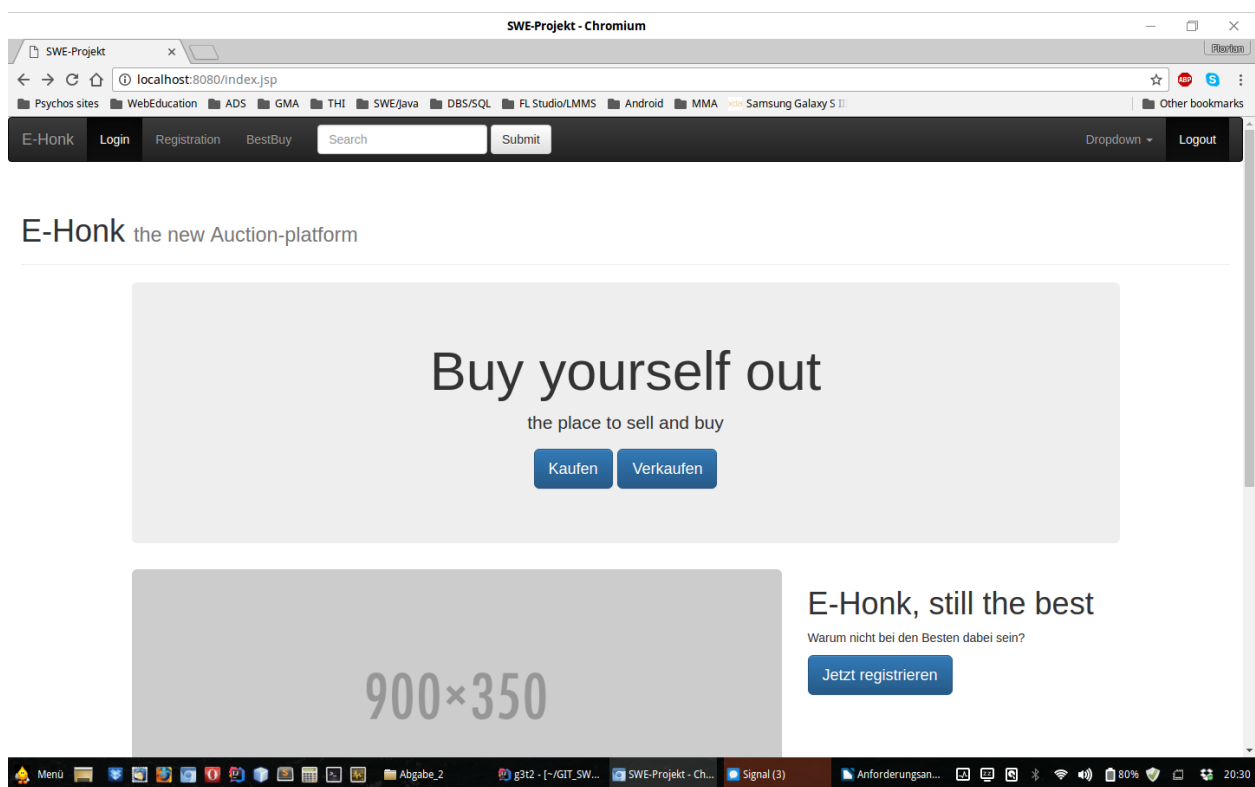
Administratoren sollen in der Lage sein neue Produktgruppen anzulegen sowie zu löschen, sowie unpassende Produkte entfernen zu können.

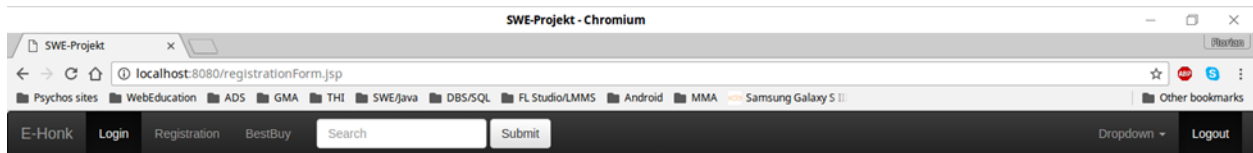
Um Forschern zu ermöglichen Informationen über das Kaufverhalten zu bekommen, muss eine Speicherung und eine Analyse der Verkäufe erfolgen.

Ebay ist eine allseits bekannte Version solch einer Plattform. Ebay unterstützt eine Vielzahl an Funktionen es finden sich nun hierzu Anleihen.

Das System soll dem User prinzipiell die Möglichkeit geben sich Artikel auszusuchen, welche auf der Plattform eingestellt wurden bzw. darauf zu bieten. Um dies zu erreichen, muss zuvor ein Login stattfinden, andernfalls können lediglich die Artikel betrachtet werden. Es soll dem User ein Überblick über die vorhandenen Artikel gegeben werden und durch Anklicken auf eine Beschreibung des Artikels verwiesen werden. Außerdem muss der User natürlich die Möglichkeit haben mittels eines einfachen Web-Formulars (nach erfolgreichem einloggen) selbst einen Artikel anbieten zu können.

1.2. Bedienoberfläche





Registration Form

First Name

Last Name

Email

Password

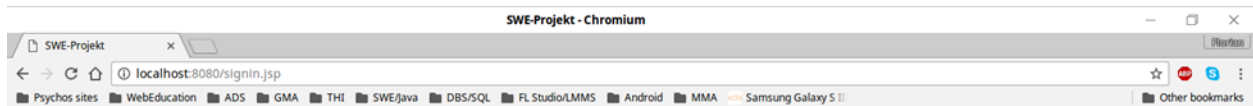
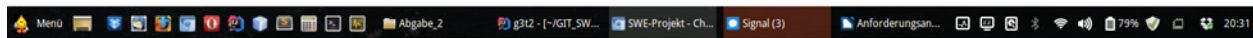
Retype Password

Date of Birth

Country

Gender ☐ Female ☐ Male ☐ Unknown

☐ I accept terms

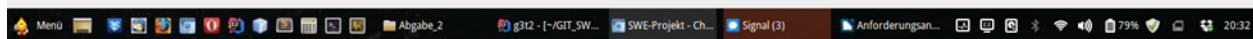


Please sign in

Email address

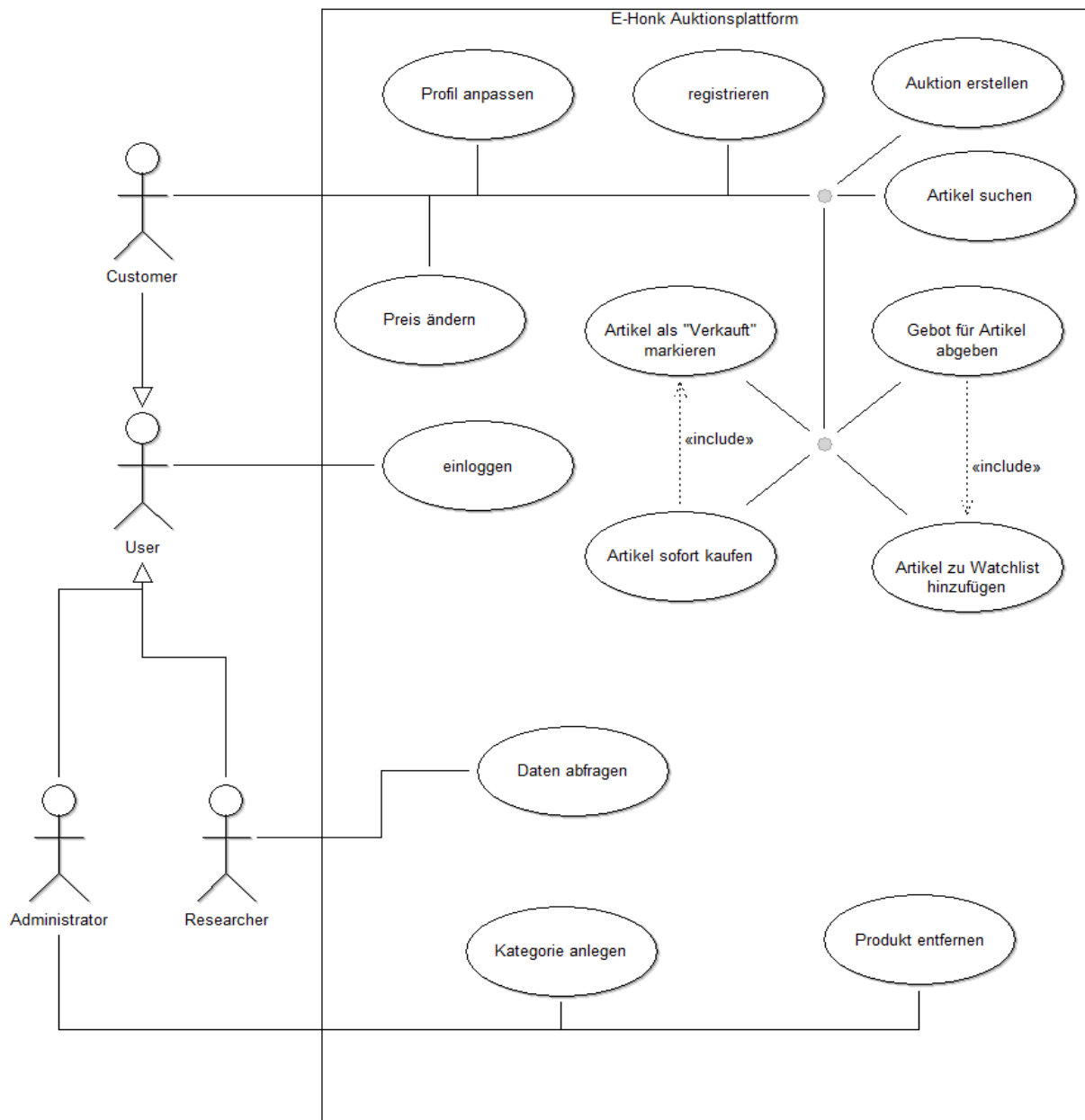
Password

☐ Remember me



1. Use-Case-Modell

1.1. Use-Case-Diagramm



1.2. Use-Case Beschreibungen

| | |
|--|---|
| <u>Use-Case 1: Registrieren</u> 1. Ziel: User-Account erstellen 2. Kurzbeschreibung: User sollen sich durch Eingabe ihrer Daten in das Anmeldeformular einen Account erstellen können 3. Vorbedingung: - 4. Nachbedingung, bei Erfolg: User besitzen nun einen persönlichen Account und können sich einloggen 5. Fehlersituationen: Fehlerhafte Eingaben, E-Mail-Adresse bereits in Verwendung 6. Nachzustand im Fehlerfall: Fehlermeldung, Daten müssen korrigiert werden um fortzufahren 7. Akteure: User 8. Trigger: Auslösendes Ereignis: - 9. Basisablauf: 1. Button „Registrieren“ anklicken 2. Formular ausfüllen 3. Button „Bestätigen“ anklicken 10. Alternativablauf: - 11. Priorität Primär | <u>Use-Case 2: Login</u> 1. Ziel: In Account einloggen 2. Kurzbeschreibung: User sollen sich durch Eingabe ihrer E-Mail-Adresse und ihres Passwortes in ihren persönlichen User-Account einloggen können 3. Vorbedingung: Use-Case 1: Registrieren 4. Nachbedingung, bei Erfolg: User können nun als Käufer oder Verkäufer agieren 5. Fehlersituationen: Fehlerhafte Eingaben, Nutzernamen oder Passwort falsch 6. Nachzustand im Fehlerfall: Fehlermeldung, Daten müssen korrigiert werden um fortzufahren 7. Akteure: User 8. Trigger: Auslösendes Ereignis: - 9. Basisablauf: 1. Button „Login“ anklicken 2. Accountdaten eingeben 3. Button „Weiter“ anklicken 10. Alternativablauf: - 11. Priorität Primär |
|--|---|

| | |
|--|--|
| <p><u>Use-Case 3: Profil anpassen</u></p> <p>1. Ziel: Persönliches Profil ändern</p> <p>2. Kurzbeschreibung: User können ihr eigenes Profil ändern / aktualisieren</p> <p>3. Vorbedingung: Use-Case 2: Login</p> <p>4. Nachbedingung, bei Erfolg: Userprofil wird aktualisiert</p> <p>5. Fehlersituationen: Fehlerhafte Eingaben</p> <p>6. Nachzustand im Fehlerfall: Fehlermeldung, Daten müssen korrigiert werden um fortzufahren</p> <p>7. Akteure: User</p> <p>8. Trigger: Auslösendes Ereignis: -</p> <p>9. Basisablauf: 1. Button „Profil“ anklicken 2. Profil ändern 3. Button „Speichern“ anklicken</p> <p>10. Alternativablauf: -</p> <p>11. Priorität Sekundär</p> | <p><u>Use-Case 4: Auktion erstellen</u></p> <p>1. Ziel: Neue Auktion erstellen</p> <p>2. Kurzbeschreibung: Verkäufer können eine neue Auktion erstellen und mittels eines Formulars alle notwendigen Daten wie Beschreibung, Titel und Preis festlegen</p> <p>3. Vorbedingung: Use-Case 2: Login</p> <p>4. Nachbedingung, bei Erfolg: Eine neue Auktion wurde erstellt und ist nun für alle User sichtbar</p> <p>5. Fehlersituationen: Fehlerhafte Eingaben</p> <p>6. Nachzustand im Fehlerfall: Fehlermeldung, Daten müssen auf zulässige Werte korrigiert werden um fortzufahren</p> <p>7. Akteure: Verkäufer</p> <p>8. Trigger: Auslösendes Ereignis: -</p> <p>9. Basisablauf: 1. Button „Neue Auktion“ anklicken 2. Formular ausfüllen eingeben 3. Button „Speichern“ anklicken</p> <p>10. Alternativablauf: -</p> <p>11. Priorität Primär</p> |
|--|--|

Use-Case 5: Preis ändern

1. Ziel:

Preis eines Artikels ändern

2. Kurzbeschreibung:

Verkäufer sollen die Höhe des Anfangspreises korrigieren können sofern noch keine Gebote für diese Auktion vorliegen

3. Vorbedingung:

Use-Case 2: Login

Use-Case 4: Auktion erstellen

4. Nachbedingung, bei Erfolg:

Auktionspreis wurde geändert

5. Fehlersituationen:

Unzulässiger Wert für den neuen Preis, bereits abgegebene Gebote vorhanden

6. Nachzustand im Fehlerfall:

Fehlermeldung, Preis muss korrigiert werden um fortzufahren / Aktion nicht möglich da Gebote vorhanden sind

7. Akteure:

Verkäufer

8. Trigger: Auslösendes Ereignis:

-

9. Basisablauf:

1. Button „Meine Auktionen“ anklicken
2. Auktion wählen
3. Button „Auktion bearbeiten“ anklicken
4. Button „Speichern“ anklicken

10. Alternativablauf:

-

11. Priorität

Sekundär

Use-Case 6: als „Verkauft“ markieren

1. Ziel:

Artikel als „Verkauft“ markieren

2. Kurzbeschreibung:

Ein Artikel wird als „Verkauft“ markiert, wenn dieser von einem Käufer über die Option „Sofortkauf“ erworben wird.

3. Vorbedingung:

Use-Case 2: Login

Use-Case 4: Auktion erstellen

4. Nachbedingung, bei Erfolg:

Auktion wurde als „Verkauft“ markiert

5. Fehlersituationen:

Artikel bereits verkauft

6. Nachzustand im Fehlerfall:

Fehlermeldung, Aktion wird abgebrochen

7. Akteure:

Verkäufer

8. Trigger: Auslösendes Ereignis:

Use-Case 7: Artikel sofort kaufen

9. Basisablauf:

1. Button „Meine Auktionen“ anklicken
2. Auktion wählen
3. Button „Auktion bearbeiten“ anklicken
4. Button „Speichern“ anklicken

10. Alternativablauf:

Verkäufer wird von Käufer kontaktiert und markiert den Artikel manuell als „Verkauft“

11. Priorität

Sekundär

Use-Case 7: Artikel sofort kaufen

1. Ziel:

Artikel sofort, ohne Auktion, kaufen

2. Kurzbeschreibung:

Käufer können einen Artikel sofort kaufen sofern der Verkäufer einen Sofortkaufspreis festgelegt hat und keine Gebote vorliegen

3. Vorbedingung:

Use-Case 2: Login

4. Nachbedingung, bei Erfolg:

Artikel wurde direkt erworben, Auktion wird als „Verkauft“ markiert

5. Fehlersituationen:

Gebote für Auktion vorhanden, Artikel bereits verkauft

6. Nachzustand im Fehlerfall:

Fehlermeldung, Aktion nicht möglich da Gebote vorhanden sind / Artikel bereits verkauft wurde

7. Akteure:

Käufer

8. Trigger: Auslösendes Ereignis:

-

9. Basisablauf:

1. Auktion auswählen
2. Auf „Sofortkauf“ klicken
3. Sofortkauf bestätigen

10. Alternativablauf:

-

11. Priorität

Sekundär

Use-Case 8: Gebot für Artikel abgeben

1. Ziel:

Auf Auktion bieten

2. Kurzbeschreibung:

Käufer kann ein Gebot für eine laufende Auktion abgeben und die Höhe des Gebots festlegen

3. Vorbedingung:

Use-Case 2: Login

4. Nachbedingung, bei Erfolg:

Aktuelles Höchstgebot der Auktion wird geändert, Artikel wird zur Watchlist des Käufers hinzugefügt (Use-Case 9: Artikel zur Watchlist hinzufügen)

5. Fehlersituationen:

Gebot niedriger als aktuelles Höchstgebot der Auktion

6. Nachzustand im Fehlerfall:

Fehlermeldung, Gebot muss korrigiert werden um fortzufahren

7. Akteure:

Käufer

8. Trigger: Auslösendes Ereignis:

-

9. Basisablauf:

1. Auktion wählen
2. Höhe des Gebots eingeben
3. Gebot bestätigen

10. Alternativablauf:

-

11. Priorität

Primär

Use-Case 9: Artikel zu Watchlist hinzufügen**1. Ziel:**

Auktion wird auf die persönliche Watchlist gesetzt

2. Kurzbeschreibung:

Die Watchlist dient dazu Auktionen welche den Käufer interessieren, oder für welche ein Gebot abgegeben wurde, zu verfolgen

3. Vorbedingung:

Use-Case 2: Login
(Use-Case 8: Gebot für Artikel abgeben)

4. Nachbedingung, bei Erfolg:

Auktion scheint in der Watchlist auf

5. Fehlersituationen:

Auktion ist bereits in der Watchlist

6. Nachzustand im Fehlerfall:

Fehlermeldung, Aktion wird abgebrochen

7. Akteure:

Käufer

8. Trigger: Auslösendes Ereignis:

Use-Case 8: Gebot für Artikel abgeben

9. Basisablauf:

1. Auktion wählen
2. Button „Zur Watchlist

hinzufügen“ anklicken

10. Alternativablauf:

Use-Case 8: Gebot für Artikel abgeben

11. Priorität

Sekundär

Use-Case 10: Produkt entfernen**1. Ziel:**

Unpassende Auktion löschen

2. Kurzbeschreibung:

Admins können unpassende oder den Richtlinien nicht entsprechende Auktionen löschen

3. Vorbedingung:**4. Nachbedingung, bei Erfolg:**

Auktion wurde entfernt

5. Fehlersituationen:

-

6. Nachzustand im Fehlerfall:

-

7. Akteure:

Administrator

8. Trigger: Auslösendes Ereignis:

-

9. Basisablauf:

1. Auktion wählen
2. Button „Auktion entfernen“ anklicken
3. Button „Bestätigen“ anklicken

10. Alternativablauf:

-

11. Priorität

Primär

| | |
|---|--|
| <p><u>Use-Case 11: Kategorie anlegen</u></p> <p>1. Ziel: Neue Artikel-Kategorie anlegen</p> <p>2. Kurzbeschreibung: Es wird eine neue Kategorie erstellt welche Verkäufer bei der Erstellung neuer Auktionen auswählen können</p> <p>3. Vorbedingung:</p> <p>4. Nachbedingung, bei Erfolg: Neue Kategorie hinzugefügt</p> <p>5. Fehlersituationen: Kategorie ist bereits vorhanden</p> <p>6. Nachzustand im Fehlerfall: Fehlermeldung, Aktion wird abgebrochen</p> <p>7. Akteure: Administrator</p> <p>8. Trigger: Auslösendes Ereignis: -</p> <p>9. Basisablauf: 1. Button „Kategorie hinzufügen“ anklicken 2. Kategorie eingeben 3. Button „Speichern“ anklicken</p> <p>10. Alternativablauf: -</p> <p>11. Priorität Primär</p> | <p><u>Use-Case 12: Daten abfragen</u></p> <p>1. Ziel: Nutzerdaten werden abgefragt</p> <p>2. Kurzbeschreibung: Forscher erhalten die Möglichkeit gewisse Daten bezüglich der Kaufverhaltens der User abzufragen.</p> <p>3. Vorbedingung: -</p> <p>4. Nachbedingung, bei Erfolg: Userdaten wurden abgefragt</p> <p>5. Fehlersituationen: -</p> <p>6. Nachzustand im Fehlerfall: -</p> <p>7. Akteure: Forscher</p> <p>8. Trigger: Auslösendes Ereignis: -</p> <p>9. Basisablauf: 1. Button „Nutzerdaten abfragen“ anklicken</p> <p>10. Alternativablauf: -</p> <p>11. Priorität Sekundär</p> |
|---|--|

2. Nichtfunktionale Anforderungen

Der Benutzer soll sich mit der Webseite und ihrer Bedienung sofort vertraut fühlen, das Layout sollte möglichst simpel gehalten werden und schnell verständlich sein. Bootstrap bietet momentan aktuelle und bekannte/vertraute Elemente. Das Aussehen soll dazu beitragen, die Zeit die ein User braucht, um einen Artikel zu suchen bzw. um einen Artikel einzustellen, kurz zu halten. Dies erhöht möglicherweise die Motivation der User, von der Webseite Gebrauch zu machen. Die Zuverlässigkeit des Systems sein soll gegeben sein, um dem User ein möglichst sicheres und beständiges Erlebnis zu bieten. Die Wachtlist soll zum klaren Überblick ebenfalls positiv beitragen.

1.1. Qualitätsanforderungen

➤ Effizienz

Das System soll ressourcenschonend sein. Laufzeit-Effizienz sowie Speicher-Effizienz wird bestmöglich implementiert.

➤ Benutzerfreundlichkeit (Usability)

Der Umgang der Web-Plattform soll einfach und beständig gewährleistet sein. Registrieren, einloggen, ersteigern und einen neuen Artikel einstellen soll mit wenigen Klicks möglich sein. Ein weiterer Beitrag zur Steigerung der Benutzerfreundlichkeit soll die Kategorisierung der Artikel sein, was eine leichtere und verständlichere Interaktion zwischen dem User und dem System ermöglicht.

➤ Zuverlässigkeit

Um eine Zuverlässigkeit des Systems zu gewähren werden Fehler bereits in der Implementierungsphase versucht abgefangen zu werden. Des Weiteren soll als auch die Plattform ständig erreichbar sein und der Wahrscheinlichkeit von Systemfehlern hintangehalten werden.

➤ Sicherheit

Datensicherheit stellt bei der Webapplikation eine zentrale Rolle dar, denn sobald sich jemand registriert, ist die Speicherung der Benutzerdaten erforderlich.

1.2. Technische Anforderungen

- **Betriebssystem:** UNIX (almighty), Mac/Windows/Linux (eigene Rechner)
- **Java Version/JDK :** 8.x
- **Formate:** XML
- **Browser :** (Safari, Firefox, Opera, Chrome)

1.3. Realisierungsanforderungen

- **Dokumentation:** Homepage
- **Installierung:** Tomcat, IntelliJ, JDK, JSP (Java-Servlet), Serialization

1.4. Diverses

Risiken:

- Unbefugter Datenzugriff
- Eine Auktion wird nicht durchgeführt
- Eingabe von falschen Daten
- Implementierungsfehler

Klassendesign

Interface-DAO:

Hier werden die prinzipiellen Methoden, welche mit der Datenbank/Serialisierung zusammenarbeiten geboten. Es stehen die Klassiker zur Verfügung, namentlich add, delete, und die get-Methoden für Artikel und User.

EntityManager:

Diese Klasse enthält den Großteil der Logik, login, createUser und Product, sowie Category, sowie die get-Methoden für User und Product. Weitere logisch geforderte Methoden können ebenfalls implementiert werden.

User:

Diese Klasse enthält alle Attribute die für den User nötig sind um sich zu registrieren und damit er einen Artikel kaufen kann. Customer haben als Instanzvariable zusätzlich eine watchlist für ihre laufenden Auktionen sowie, im gegebenen Fall auch ihre zu verfolgenden Artikel.

Admin:

Die Klasse Admin wird aus User abgeleitet.

Researcher:

Diese Klasse enthält zusätzlich ein Daten-Array um die Verkäufe nach verschiedenen Gesichtspunkten analysieren zu können.

Product:

Die Klasse enthält alle nötigen Instanzvariablen für die Artikel. Diese sind namentlich ID, title, category, description, sofortkauf/instantbuy, aktueller Preis, Startpreis, die VerkäuferID, die ID des Bieters mit dem höchsten Gebot und ein Start- sowie End-Datum.

Category:

Die Klasse enthält einen Namen und eine Beschreibung.

