Francesco Vitolo  
 WI20Z1A

Pflichtenheft

Pflichtenheft

Inhalt

[1. Ausgangssituation 2](#_Toc92834349)

[2. Zielsetzung 2](#_Toc92834350)

[3. Produktumgebung/Schnittstellenübersicht 2](#_Toc92834351)

[4. Muss Kriterien 3](#_Toc92834352)

[5. Kann - Kriterien 3](#_Toc92834353)

[6. Projektplan 4](#_Toc92834354)

[7. Use - Case - Diagramme 5](#_Toc92834356)

[8. GUI 7](#_Toc92834357)

[9. ER - Diagramm 10](#_Toc92834369)

[10. Git - Repository 10](#_Toc92834370)

[11. Testplan 11](#_Toc92834371)

# Ausgangssituation

Vinothek mit Fokus auf italienischen Weinen aller Art und regelmäßig wechselnder Produktpalette. Die Anzahl der unterschiedlichen Weine beläuft sich auf mehrere hunderte. Keine effiziente, robuste Möglichkeit die Weine und Produzenten zu erfassen und zu verwalten.

Das Beschaffen und das Erfassen von Informationen der einzelnen Produkte ist ein langer, mühevoller Prozess und wird deshalb immer wieder vernachlässigt. Das Eintragen in Excel Tabellen oder auf dem Papier ist über einen langen Zeitraum keine gute Lösung und sehr fehleranfällig.

Die Planung von Events (z.B. Weinverkostungen) erfolgt nur auf Papier und ist unkonkret und unpräzise formuliert. Weinverkostungen finden in der Regel jedes Wochenende, mit bis zu 20 Personen, statt. Jedes Mal werden andere Produkte vorgestellt.

# Zielsetzung

* Ziel ist es, die Informationen über die einzelnen Weine und Produzenten mit dem Computer zu erfassen und zu verwalten.
* ~~Es sollen nur italienische Weine verwaltet werden~~
* Dieses Programm soll alle Informationen an einem Ort zusammenführen, um eine detaillierte Übersicht über die Produkte und ihrer Produzenten zu liefern
* Das Erstellen, Löschen und Bearbeiten von Weinen muss möglich sein
* Das Erstellen, Löschen und Bearbeiten von Produzenten muss möglich sein
* Automatisches Erstellen von Infoblättern
* Planung von Veranstaltungen mit Termin und eingesetzten Weinen

# Produktumgebung/Schnittstellenübersicht

* Grafische Benutzeroberfläche
* Datenbankanbindung
* PDF - Dateien erstellen und speichern
* Betriebssystem Windows 10
* WPF
* xaml
* C#
* .NET Framework 4.7.2
* Entity Framework
* SQL
* PDFsharp

# Muss Kriterien

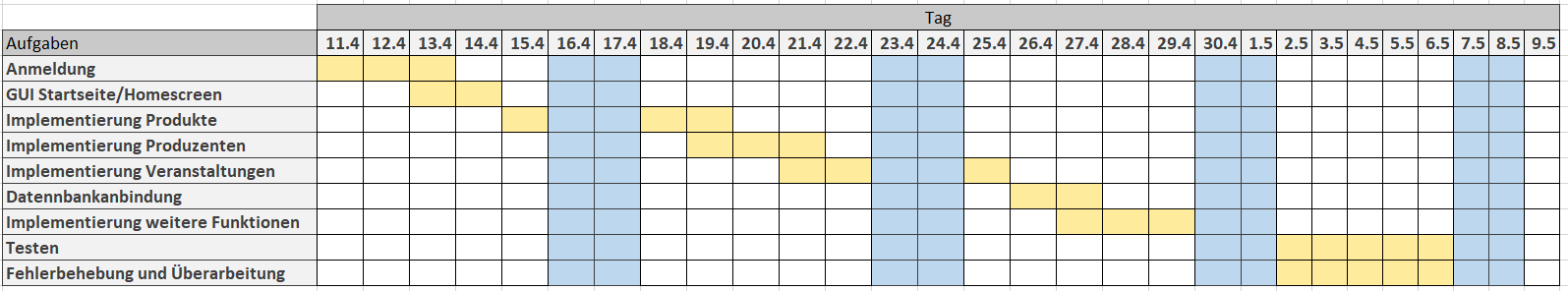
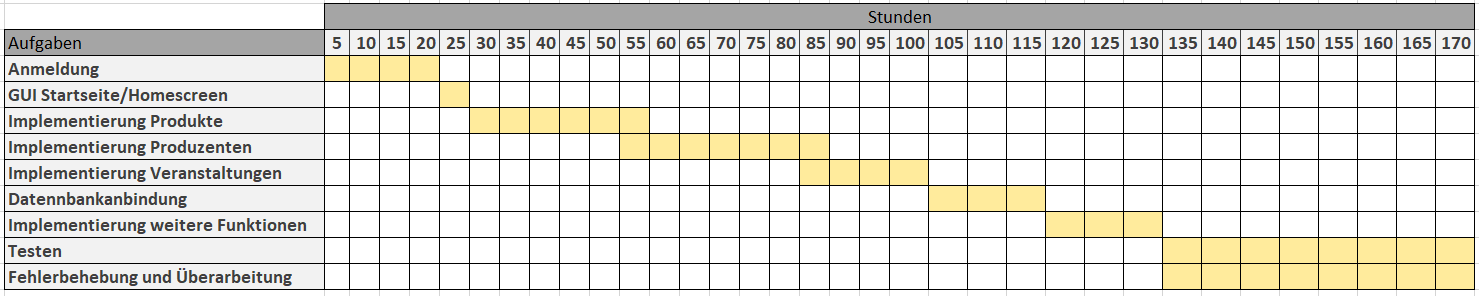
* Das System ermöglicht eine Anmeldung mit Nutzername und Passwort (20 Stunden)
* Startseite erstellen (5 Stunden)
* Weine sollen hinzugefügt werden können (5 Stunden)
* Weine sollen bearbeitet werden können (5 Stunden)
* Weine sollen gelöscht werden können (5 Stunden)
* Liste mit Weinen inklusive Suchfunktion (10 Stunden)
* Filterfunktion für Weine (5 Stunden)
* Produzent hinzufügen (5 Stunden)
* Produzent entfernen (5 Stunden)
* Produzent bearbeiten (5 Stunden)
* Liste mit Produzenten inklusive Suchfunktion (5 Stunden)
* Filterfunktion für Produzenten (10 Stunden)
* Veranstaltung hinzufügen (5 Stunden)
* Veranstaltung entfernen (5 Stunden)
* Veranstaltung bearbeiten (5 Stunden)
* Liste mit Veranstaltungen (5 Stunden)
* Filterfunktion Veranstaltungen (10 Stunden)

# Kann - Kriterien

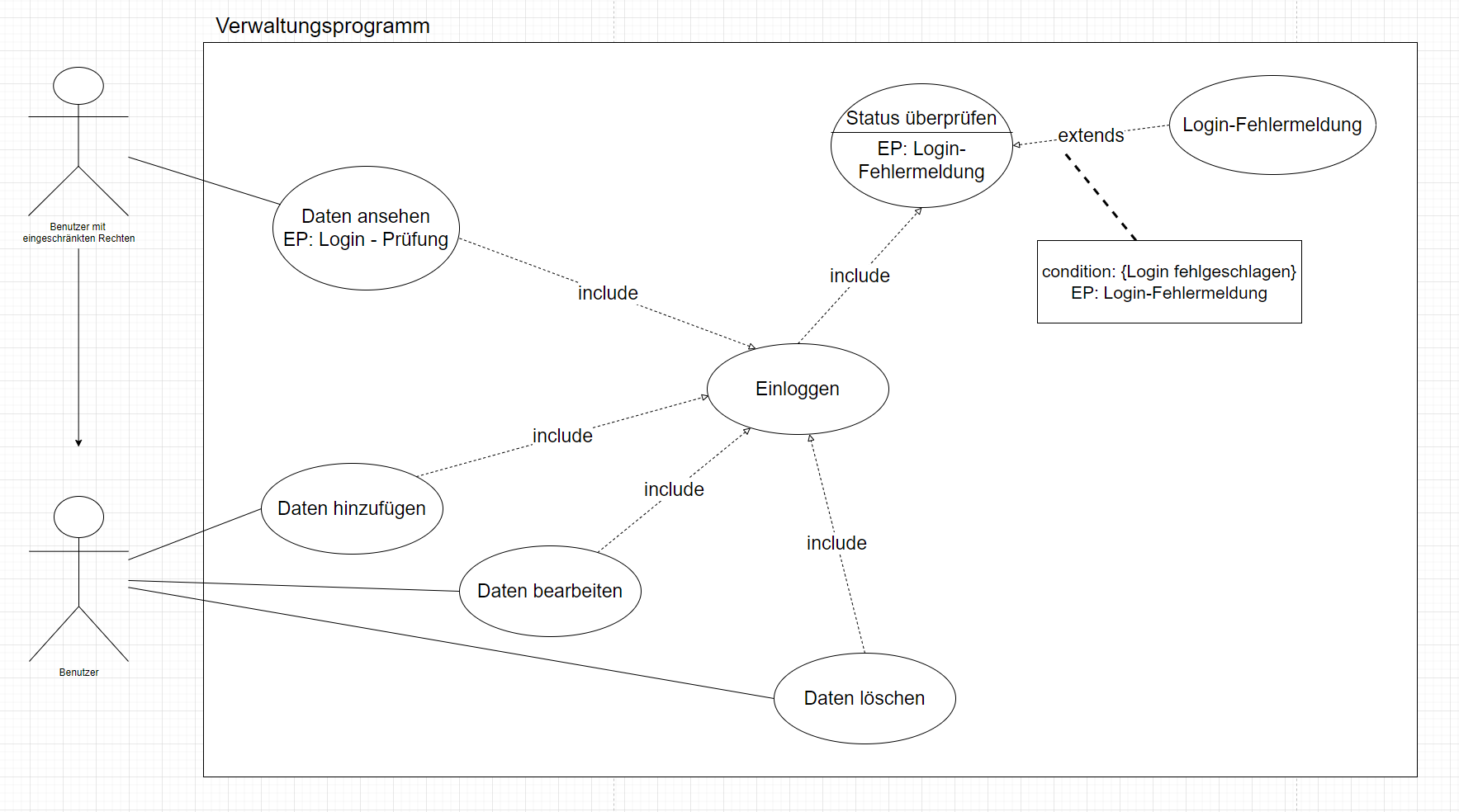
* Eine „Lite“ Version für Kunden (10 Stunden)
* Erstellen von Informationsblättern zu einzelnen Weinen (10 Stunden)
* ~~Auswahl des/der Wein(e) des Monats (5 Stunden)~~
* Der Nutzer kann zwischen Dunkelmodus und Hellmodus wechseln (5 Stunden)

# Projektplan

## Gantt – Diagramme

# Use - Case - Diagramme



|  |  |
| --- | --- |
| **Use-Case Name:** | **Nutzer meldet sich an** |
| **Kurzbeschreibung:** | Ein Benutzer möchte sich anmelden |
| **Vorbedingung:** | Das Programm wurde gestartet |
| **Nachbedingung:** | Die Funktionen des Programms sind nutzbar |
| **Fehlersituation:** | Falsche Login - Daten |
| **Systemzustand im Fehlerfall:** | Das Programm bleibt unverändert |
| **Akteure:** | Benutzer, Programm |
| **Trigger:** | Ein Benutzer möchte sich anmelden |
| **Standardablauf:** | * + 1. Nutzer startet das Programm     2. Nutzer wird aufgefordert Daten einzugeben     3. Nutzer bestätigt die Eingabe     4. Nutzer wird angemeldet |
| **Alternativabläufe:** | 4‘ Falsche Anmeldedaten |

|  |  |
| --- | --- |
| **Use-Case Name:** | **Event löschen** |
| **Kurzbeschreibung:** | Ein Benutzer möchte eine Veranstaltung löschen. |
| **Vorbedingung:** | Die Person ist angemeldet. |
| **Nachbedingung:** | Es wird eine Veranstaltung gelöscht. |
| **Fehlersituation:** | Die Veranstaltung liegt in der Vergangenheit |
| **Systemzustand im Fehlerfall:** | Die Datenbank bleibt unverändert |
| **Akteure:** | Benutzer, Datenbank |
| **Trigger:** | Benutzer möchte eine alte Veranstaltung entfernen |
| **Standardablauf:** | 1. Nutzer wählt die gewollte Veranstaltung aus. 2. Nutzer wählt die Funktion „Veranstaltung löschen“. 3. Nutzer wird aufgefordert die Aktion zu bestätigen. 4. Nutzer wählt „bestätigen“. 5. Veranstaltung wird gelöscht |
| **Alternativabläufe:** | 3‘ Die Veranstaltung liegt in der Vergangenheit |

|  |  |
| --- | --- |
| **Use-Case Name:** | **Wein hinzufügen** |
| **Kurzbeschreibung:** | Ein Benutzer fügt einen Wein hinzu. |
| **Vorbedingung:** | Die Person ist angemeldet. |
| **Nachbedingung:** | Es wird ein Wein hinzugefügt. |
| **Fehlersituation:** | Das Produkt ist schon vorhanden. |
| **Systemzustand im Fehlerfall:** | Die Datenbank bleibt unverändert. |
| **Akteure:** | Benutzer, Datenbank. |
| **Trigger:** | Benutzer möchte Wein hinzufügen |
| **Standardablauf:** | 1. Nutzer wählt die Funktion „Wein hinzufügen“. 2. Nutzer wird aufgefordert Daten einzugeben. 3. Nutzer wählt „speichern“ aus. 4. Produkt wird hinzugefügt. |
| **Alternativabläufe:** | 4‘ Produkt schon vorhanden. |

# GUI

## Anmelden

## 

Bedingung: falsche Login - Daten

## Startbildschirm



*Dunkelmodus umschalten*

*Dunkelmodus umschalten*

## Produkt hinzufügen/ändern/ansehen (3 Fenster)

Button PDF – Infoblatt erstellen (optional)

## 

Button

Bild

Button\_Speichern

Beschreibung

…

…

…

Name:

…

…

…

Button

Button\_Umschalten\_bearbeiten/ansehen

## Produzent hinzufügen/ändern/ansehen (3 Fenster)

## 

Button

Button\_Umschalten\_bearbeiten/ansehen

Informationen

…

…

Name: …

…

…

…

Button\_Speichern

## Veranstaltung hinzufügen/ändern/ansehen (3 Fenster)

## 

Liste Weine

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |

Scrollbar

Name:

…

…

…

Button\_Speichern

Button

Button\_Umschalten\_bearbeiten/ansehen

Button\_Löschen

## Übersicht Produkte/Produzenten/Veranstaltungen (3 Fenster)

## 

Suchleiste

Übersicht Produkte/Produzenten/Veranstaltungen

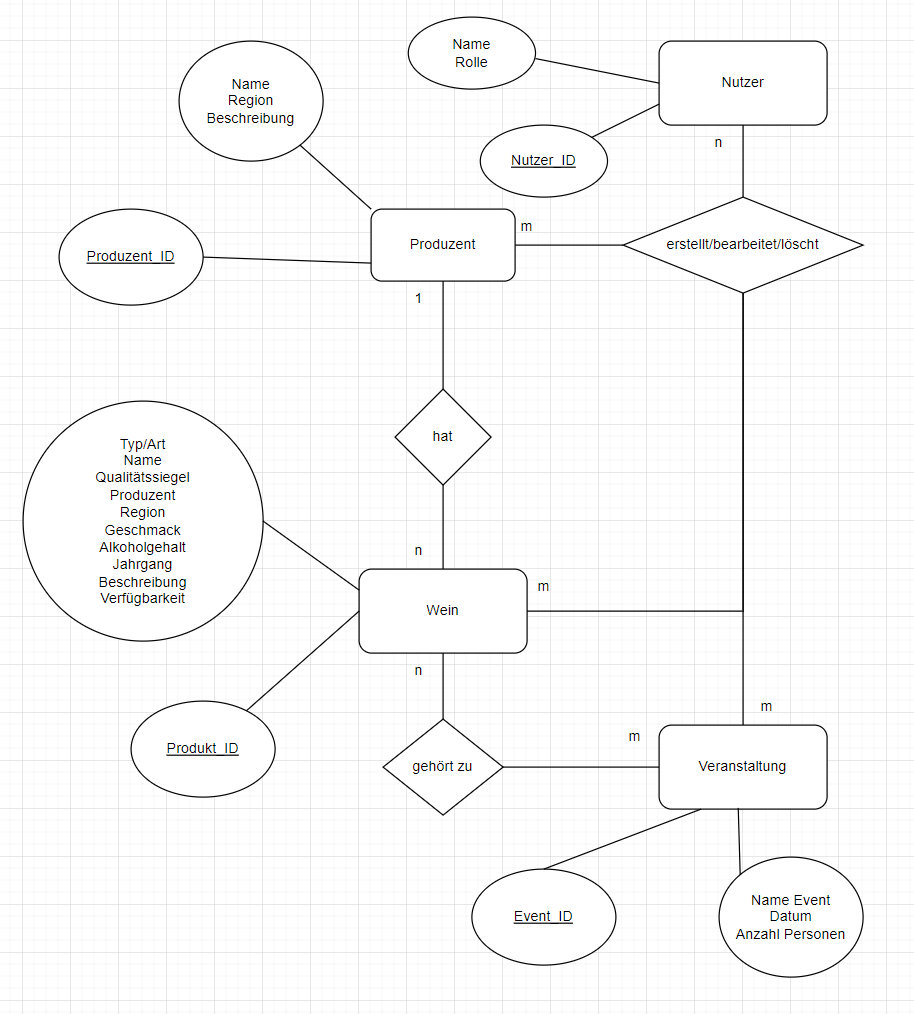
+

X

Button\_Ändern

Button\_Hinzufügen

# ER - Diagramm



1

1

1

# Git - Repository

<https://github.com/Francesco-Vitolo/Hausarbeit>

# Testplan

## Allgemein

| **ID:** |  |
| --- | --- |
| **Beschreibung:** | Einloggen als Benutzer. |
| **Vorbedingung:** | Das Programm wird gestartet. |
| **Test-Schritte:** | 1. Im Feld “Name” wird der Name eines Accounts eingegeben. |
|  | 2. Im Feld “Passwort” wird das passende Passwort eingegeben. |
|  | 3. Der Login wird durchgeführt. |
| **Erwartetes Resultat:** | Der Nutzer ist eingeloggt. |

| **ID:** |  |
| --- | --- |
| **Beschreibung:** | Informationen ändern |
| **Vorbedingung:** | Nutzer ist eingeloggt |
| **Test-Schritte:** | 1. Es wird die zu ändernde Position ausgewählt.  1‘. Es wird nach dem Objekt durch Eingabe in die Suchleiste gesucht. |
|  | 2. Der Nutzer wählt das Objekt durch Doppelklick aus. |
|  | 3. Es wird das Fenster mit den Informationen geöffnet.  4. Der Button „bearbeiten“ wird gedrückt.  5. Die gewünschten Änderungen werden eingetragen  6. Die Änderungen werden durch das Feld „Sichern“ übernommen |
| **Erwartetes Resultat:** | Der Nutzer hat Eigenschaften geändert |

| **ID:** |  |
| --- | --- |
| **Beschreibung:** | Hintergrund/Schriftfarbe/Buttonfarbe ändern |
| **Vorbedingung:** | Nutzer ist eingeloggt. |
| **Test-Schritte:** | 1. Es wird in die Einstellungen navigiert. 1‘. Es wird durch Drücken auf den Button „Googles Color Picker“ die gewünschte Farbe ausgewählt und der Code wurde kopiert. |
|  | 2. Es wird der Hexadezimalcode der Farbe in das gewünschte Feld eingetragen. |
|  | 3. Die Änderungen werden durch Drücken des Buttons übernommen. |
| **Erwartetes Resultat:** | Die Farbe wurde dauerhaft geändert. |

| **ID:** |  |
| --- | --- |
| **Beschreibung:** | Das Standardverzeichnis für die Infoblätter wird geändert. |
| **Vorbedingung:** | Nutzer ist eingeloggt. |
| **Test-Schritte:** | 1. Es wird in die Einstellungen navigiert.  2. Durch Drücken auf das Verzeichnissymbol kann das Verzeichnis geändert werden.  3. Durch Bestätigung werden die Änderungen übernommen |
| **Erwartetes Resultat:** | Das Verzeichnis wurde dauerhaft geändert |

| **ID:** |  |
| --- | --- |
| **Beschreibung:** | Nutzer löschen |
| **Vorbedingung:** | Nutzer ist als Admin eingeloggt. |
| **Test-Schritte:** | 1. Es wird in die Einstellungen navigiert.  2. Durch Drücken auf das Nutzersymbol öffnet sich die Nutzerverwaltung  3. Der Nutzer wird ausgewählt  4. Der Button „Nutzer löschen“ wird gedrückt.  5. Die Aktion wird bestätigt |
| **Erwartetes Resultat:** | Der Nutzer wurde gelöscht |

| **ID:** |  |
| --- | --- |
| **Beschreibung:** | Passwort ändern |
| **Vorbedingung:** | Nutzer ist als Admin eingeloggt. |
| **Test-Schritte:** | 1. Es wird in die Einstellungen navigiert.  2. Durch Drücken auf das Nutzersymbol öffnet sich die Nutzerverwaltung.  3. Der Nutzer wird ausgewählt.  4. Das neue Passwort wird eingetragen.  5. Der Button „Passwort ändern“ wird gedrückt. |
| **Erwartetes Resultat:** | Das Passwort wurde geändert |

| **ID:** |  |
| --- | --- |
| **Beschreibung:** | Nutzer hinzufügen |
| **Vorbedingung:** | Nutzer ist als Admin eingeloggt. |
| **Test-Schritte:** | 1. Es wird in die Einstellungen navigiert.  2. Durch Drücken auf das Nutzersymbol öffnet sich die Nutzerverwaltung.  3. Der Name des Nutzers wird eingetragen.  4. Das neue Passwort wird eingetragen.  5. Der Button „Nutzer erstellen“ wird gedrückt. |
| **Erwartetes Resultat:** | Ein neuer Nutzer wurde hinzugefügt. |

| **ID:** |  |
| --- | --- |
| **Beschreibung:** | Logins überprüfen |
| **Vorbedingung:** | Nutzer ist als Admin eingeloggt. |
| **Test-Schritte:** | 1. Es wird in die Einstellungen navigiert.  2. Durch Drücken auf das Listensymbol öffnet sich die Übersicht.  3. Eine Liste mit allen Anmeldungen öffnet sich. |
| **Erwartetes Resultat:** | Der Admin kann überprüfen, wer sich eingeloggt hat. |

## Produkte

| **ID:** |  |
| --- | --- |
| **Beschreibung:** | Alle Produkte anzeigen. |
| **Vorbedingung:** | Der Nutzer befindet sich im Hauptmenü. |
| **Test-Schritte:** | 1. Es wird das Feld Übersicht der Produkte ausgewählt. |
|  | 2. Das Fenster „Liste Produkte“ öffnet sich. |
| **Erwartetes Resultat:** | Der Nutzer hat eine Liste aller angelegten Produkte. |

| **ID:** |  |
| --- | --- |
| **Beschreibung:** | Produkt hinzufügen. |
| **Vorbedingung:** | Es wird auf den Button „Produkt hinzufügen“ gedrückt. |
| **Test-Schritte:** | 1. Die Informationen werden eingetragen. |
|  | 2. Ein Bild wird ausgewählt. |
|  | 2‘. Ein Bild wird aus der Zwischenablage eingefügt.  2‘‘. Es wird kein Bild hinzugefügt.  3. Es wird auf den Button „Sichern“ gedrückt.  4. Es wird in die Datenbank übernommen. |
| **Erwartetes Resultat:** | Ein neues Produkt wurde hinzugefügt |

| **ID:** |  |
| --- | --- |
| **Beschreibung:** | Produkt löschen. |
| **Vorbedingung:** | Das Programm ist im Dialog „Liste aller Produkte“. |
| **Test-Schritte:** | 1. Es wird die zu löschende Position ausgewählt.  1‘. Es wird nach dem Produkt durch Eingabe in der Suchleiste gesucht. |
|  | 2. Der Nutzer drückt auf den Button „löschen“. |
|  | 3. Es wird der Bestätigungsdialog geöffnet.  4. Die Aktion wird bestätigt.  5. Ein Produkt wird gelöscht. |
| **Erwartetes Resultat:** | Der Nutzer hat das Produkt gelöscht. |

| **ID:** |  |
| --- | --- |
| **Beschreibung:** | Produkt duplizieren. |
| **Vorbedingung:** | Das Programm ist im Dialog „Liste aller Produkte“. |
| **Test-Schritte:** | 1. Es wird die zu kopierende Position ausgewählt.  1‘. Es wird nach dem Produkt durch Eingabe in der Suchleiste gesucht. |
|  | 2. Der Nutzer drückt auf den Button „duplizieren“. |
|  | 3. Das Produkt wird dupliziert. |
| **Erwartetes Resultat:** | Der Nutzer hat ein Produkt mit den gleichen Informationen erschaffen. |

| **ID:** |  |
| --- | --- |
| **Beschreibung:** | Produkt suchen. |
| **Vorbedingung:** | Das Programm ist im Dialog „Liste aller Produkte“. |
| **Test-Schritte:** | 1. Es wird auf die Suchleiste gedrückt  2. Es wird das gewünschte Produkt eingegeben  3. Das Programm zeigt passende Ergebnisse an. |
|  |  |
| **Erwartetes Resultat:** | Der Nutzer hat das gewünschte Produkt gefunden. |

| **ID:** |  |
| --- | --- |
| **Beschreibung:** | Sortieren der Produkte |
| **Vorbedingung:** | Das Programm ist im Dialog „Liste aller Produkte“. |
| **Test-Schritte:** | 1. Es wird die gewünschte Eigenschaft ausgewählt  1‘. Die Box absteigend/aufsteigend wird gedrückt  2. Das Programm sortiert nach der gewünschten Eigenschaft |
|  |  |
| **Erwartetes Resultat:** | Die Sortierreihenfolge hat sich geändert. |

| **ID:** |  |
| --- | --- |
| **Beschreibung:** | Infoblätter erstellen |
| **Vorbedingung:** | Das Programm zeigt alle Informationen an |
| **Test-Schritte:** | 1. Es wird auf den „Button PDF – Datei erstellen“ gedrückt.  2. Eine PDF – Datei wird erstellt.  3. Die PDF – Datei wird im Standard – Verzeichnis gespeichert. |
|  |  |
| **Erwartetes Resultat:** | Ein Infoblatt wurde erstellt und in der Datenbank gespeichert |

| **ID:** |  |
| --- | --- |
| **Beschreibung:** | Infoblätter anschauen |
| **Vorbedingung:** | Ein Infoblatt wurde erstellt. |
| **Test-Schritte:** | 1. Es wird auf den „Button PDF – Datei anschauen“ gedrückt.  2. Eine PDF – Datei wird angezeigt. |
| **Erwartetes Resultat:** | Die PDF -Datei wird angezeigt. |

| **ID:** |  |
| --- | --- |
| **Beschreibung:** | Infoblätter erneut herunterladen |
| **Vorbedingung:** | Ein Infoblatt wurde erstellt. |
| **Test-Schritte:** | 1. Es wird auf den „Button PDF – Datei herunterladen“ gedrückt.  2. Der Dialog für der Auswahl des Speicherorts wird angezeigt.  3. Der Nutzer wählt den Speicherort aus und bestätigt seine Eingabe |
| **Erwartetes Resultat:** | Die PDF -Datei wurde heruntergeladen. |

| **ID:** |  |
| --- | --- |
| **Beschreibung:** | Websuche |
| **Vorbedingung:** | Die Seite mit den Informationen zum Produkt wird angezeigt. |
| **Test-Schritte:** | 1. Es wird auf den Button „Websuche“ gedrückt.  2. Der Standard – Browser öffnet sich.  3. Es werden Suchergebnisse zum Namen des Produkts und zum Weingut angezeigt. |
| **Erwartetes Resultat:** | Es wurde eine Websuche durchgeführt. |

## Produzenten

| **ID:** |  |
| --- | --- |
| **Beschreibung:** | Alle Produzenten anzeigen. |
| **Vorbedingung:** | Der Nutzer befindet sich im Hauptmenü. |
| **Test-Schritte:** | 1. Es wird das Feld „Übersicht Produzenten“ ausgewählt. |
|  | 2. Das Fenster öffnet sich. |
| **Erwartetes Resultat:** | Der Nutzer hat eine Liste aller angelegten Produzenten. |

| **ID:** |  |
| --- | --- |
| **Beschreibung:** | Produzent hinzufügen. |
| **Vorbedingung:** | Es wird auf den Button „Produzent hinzufügen“ gedrückt. |
| **Test-Schritte:** | 1. Die Informationen werden eingetragen. |
|  | 2. Es wird auf den Button „Sichern“ gedrückt.  3. Es wird in die Datenbank übernommen. |
| **Erwartetes Resultat:** | Ein neuer Produzent wurde hinzugefügt |

| **ID:** |  |
| --- | --- |
| **Beschreibung:** | Produzent löschen. |
| **Vorbedingung:** | Das Programm ist im Dialog „Liste aller Produzenten“. |
| **Test-Schritte:** | 1. Es wird die zu löschende Position ausgewählt.  1‘. Es wird nach dem Produzenten durch Eingabe in der Suchleiste gesucht. |
|  | 2. Der Nutzer drückt auf den Button „löschen“. |
|  | 3. Es wird der Bestätigungsdialog geöffnet.  4. Die Aktion wird bestätigt.  5. Ein Produzent wird gelöscht. |
| **Erwartetes Resultat:** | Der Nutzer hat den Produzenten gelöscht. |

| **ID:** |  |
| --- | --- |
| **Beschreibung:** | Produzent duplizieren. |
| **Vorbedingung:** | Das Programm ist im Dialog „Liste aller Produzenten“. |
| **Test-Schritte:** | 1. Es wird die zu kopierende Position ausgewählt.  1‘. Es wird nach dem Produzenten durch Eingabe in der Suchleiste gesucht. |
|  | 2. Der Nutzer drückt auf den Button „duplizieren“. |
|  | 3. Der Produzent wird dupliziert. |
| **Erwartetes Resultat:** | Der Nutzer hat einen Produzenten mit den gleichen Informationen erschaffen. |

| **ID:** |  |
| --- | --- |
| **Beschreibung:** | Produzent suchen. |
| **Vorbedingung:** | Das Programm ist im Dialog „Liste aller Produzenten“. |
| **Test-Schritte:** | 1. Es wird auf die Suchleiste gedrückt  2. Es wird der gewünschte Produzent eingegeben  3. Das Programm zeigt passende Ergebnisse an. |
|  |  |
| **Erwartetes Resultat:** | Der Nutzer hat den gewünschten Produzenten gefunden. |

| **ID:** |  |
| --- | --- |
| **Beschreibung:** | Sortieren der Produzenten |
| **Vorbedingung:** | Das Programm ist im Dialog „Liste aller Produzente“. |
| **Test-Schritte:** | 1. Es wird die gewünschte Eigenschaft ausgewählt  1‘. Die Box absteigend/aufsteigend wird gedrückt  2. Das Programm sortiert nach der gewünschten Eigenschaft |
|  |  |
| **Erwartetes Resultat:** | Die Sortierreihenfolge hat sich geändert. |

| **ID:** |  |
| --- | --- |
| **Beschreibung:** | Email versenden |
| **Vorbedingung:** | Email - Adresse vorhanden |
| **Test-Schritte:** | 1. Es wird auf den Button „Email versenden“ gedrückt.  2. Das Standard – Emailprogramm öffnet sich  3. Die Email wird automatisch in das Feld des Empfängers eingetragen |
| **Erwartetes Resultat:** | Der Nutzer kann eine Email versenden |

| **ID:** |  |
| --- | --- |
| **Beschreibung:** | Produkt bearbeiten in Ansicht „Produzent bearbeiten“ |
| **Vorbedingung:** | Produzent besitzt mindestens 1 Produkt |
| **Test-Schritte:** | 1. Es wird durch Doppelklick auf das gewünschte Produkt gedrückt  2. Die Ansicht „Produkt bearbeiten/ansehen“ öffnet sich  3. siehe T2  4. Es wird zurücknavigiert  5. Die Änderungen wurden in der Ansicht übernommen |
| **Erwartetes Resultat:** | Die Änderungen wurden übernommen und werden richtig angezeigt |

## Veranstaltungen

| **ID:** |  |
| --- | --- |
| **Beschreibung:** | Alle Veranstaltungen anzeigen. |
| **Vorbedingung:** | Der Nutzer befindet sich im Hauptmenü. |
| **Test-Schritte:** | 1. Es wird das Feld „Übersicht Veranstaltungen“ ausgewählt. |
|  | 2. Das Fenster öffnet sich. |
| **Erwartetes Resultat:** | Der Nutzer hat eine Liste aller angelegten Veranstaltungen. |

| **ID:** |  |
| --- | --- |
| **Beschreibung:** | Veranstaltung planen. |
| **Vorbedingung:** | Das Programm ist im Dialog „Veranstaltung hinzufügen“ |
| **Test-Schritte:** | 1. Im Feld “Name” wird der Name der Veranstaltung eingegeben. |
|  | 2. Im Feld “Datum“ wird das Datum eingegeben. |
|  | 3. Im Feld “Anzahl Personen“ wird die Anzahl der Personen, die teilnehmen, eingetragen.  4. Durch Drücken des Buttons „Wein hinzufügen“ wird der Dialog „Liste der Weine“ geöffnet.  5. Das gewünschte Produkt wird durch Doppelklick ausgewählt.  6. Ein Wein wird hinzugefügt.  7. Die Veranstaltung wird durch den Klick auf den Button „Sichern“ gespeichert.  7‘. Schritte 4 und 5 werden wiederholt. |
| **Erwartetes Resultat:** | Der Nutzer hat eine Veranstaltung geplant. |

| **ID:** |  |
| --- | --- |
| **Beschreibung:** | Veranstaltung löschen. |
| **Vorbedingung:** | Das Programm ist im Dialog „Liste aller Veranstaltungen“. |
| **Test-Schritte:** | 1. Es wird die zu löschende Position ausgewählt.  1‘. Es wird nach der Veranstaltung durch Eingabe in der Suchleiste gesucht. |
|  | 2. Der Nutzer drückt auf den Button „löschen“. |
|  | 3. Es wird der Bestätigungsdialog geöffnet.  4. Die Aktion wird bestätigt.  5. Eine Veranstaltung wird gelöscht. |
| **Erwartetes Resultat:** | Der Nutzer hat die Veranstaltung gelöscht. |

| **ID:** |  |
| --- | --- |
| **Beschreibung:** | Veranstaltung duplizieren. |
| **Vorbedingung:** | Das Programm ist im Dialog „Liste aller Veranstaltungen“. |
| **Test-Schritte:** | 1. Es wird die zu kopierende Position ausgewählt.  1‘. Es wird nach der Veranstaltung durch Eingabe in der Suchleiste gesucht. |
|  | 2. Der Nutzer drückt auf den Button „duplizieren“. |
|  | 3. Die Veranstaltung wird dupliziert. |
| **Erwartetes Resultat:** | Der Nutzer hat eine Veranstaltung mit den gleichen Informationen erschaffen. |

| **ID:** |  |
| --- | --- |
| **Beschreibung:** | Veranstaltung suchen. |
| **Vorbedingung:** | Das Programm ist im Dialog „Liste aller Veranstaltungen“. |
| **Test-Schritte:** | 1. Es wird auf die Suchleiste gedrückt  2. Es wird die gewünschte Veranstaltung eingegeben  3. Das Programm zeigt passende Ergebnisse an. |
|  |  |
| **Erwartetes Resultat:** | Der Nutzer hat die gewünschte Veranstaltung gefunden. |

| **ID:** |  |
| --- | --- |
| **Beschreibung:** | Sortieren der Veranstaltungen |
| **Vorbedingung:** | Das Programm ist im Dialog „Liste aller Veranstaltung“. |
| **Test-Schritte:** | 1. Es wird die gewünschte Eigenschaft ausgewählt  1‘. Die Box absteigend/aufsteigend wird gedrückt  2. Das Programm sortiert nach der gewünschten Eigenschaft |
|  |  |
| **Erwartetes Resultat:** | Die Sortierreihenfolge hat sich geändert. |

| **ID:** |  |
| --- | --- |
| **Beschreibung:** | Produkt bearbeiten in Ansicht „Veranstaltung planen“ |
| **Vorbedingung:** | Veranstaltung besitzt mindestens 1 Produkt |
| **Test-Schritte:** | 1. Es wird durch Doppelklick auf das gewünschte Produkt gedrückt.  2. Die Ansicht „Produkt bearbeiten/ansehen“ öffnet sich.  3. siehe T2.  4. Es wird zurücknavigiert.  5. Die Änderungen wurden in der Ansicht übernommen. |
| **Erwartetes Resultat:** | Die Änderungen wurden übernommen und werden richtig angezeigt |

|  |  |
| --- | --- |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |