

酒屋在庫管理システム

グループ名：A チーム

メンバー：大田和樹・徳富秀輔

酒屋在庫システム

作成者：大田和樹・徳富秀輔

1. 要求定義書

1.はじめに

2019/6/13 に出題された課題,「酒屋在庫管理システム(簡易バージョン)」の要求を分析し,定義するものである.

2.酒屋在庫管理システム

要求される業務システムの概要は以下図の通りである.

酒屋在庫管理システム(簡易バージョン)

- ある酒屋は, 卸売業者や顧客, 販売員の依頼に応じて, さまざまな業務を行う.
- 顧客からの注文があると, 酒の出荷を行う. 注文は, 客の名前, 酒の銘柄, 本数で指定される. 各注文には, 自動的に一意な注文番号がつけられる. 指定された銘柄が存在しない, または, 在庫がなければ酒を出荷しない. 銘柄が存在して, 在庫も十分にあれば, 出荷を行い, 在庫を更新する. 出荷の際には, 出荷伝票を顧客に発行すると共に, その控えを残す. 出荷伝票には, 注文番号, 客の名前, 酒の銘柄, 本数が記載される.
- 卸売業者からの入荷依頼が届くと, 酒を入荷する. 入荷依頼は, 酒の銘柄と本数で指定される. 倉庫内に, 既に同じ銘柄の酒があれば, 該当する銘柄の在庫本数に加算する. なければ, 新たに銘柄とその本数を在庫に追加する.
- 販売員の依頼に応じて, 現在の在庫表示を行う. 在庫表示は, 倉庫内にある全ての酒の銘柄とその本数をリストにして表示するものである.
- 販売員の依頼に応じて, 出荷実績の表示を行う. 出荷実績とは, 顧客名を与えると, その顧客に過去に出荷したすべての出荷伝票のリストを表示するものである.
- 倉庫の在庫状況, 出荷伝票控えは, ファイル(またはDB)によって保存される. これらのファイルは, システム起動時に自動的に読み込まれ, システム終了時に終了時点のデータがファイルに書き込まれるものとする.

3. 要求定義

2.の酒屋在庫管理システムを実現する「酒屋在庫管理システムプログラム」の要求を分析し定義する.

酒屋在庫管理システムプログラムは, 主に以下の 6 つの要求をすべて満たすプログラムとする.

3.1 酒を出荷

[send1:]顧客からの注文を受ける

[send1.1:]客の名前

[send1.2:]酒の銘柄

[send1.3:]本数

[send2:]注文番号をつける

[send2.1:]在庫があるかどうか確認する

[send2.2:]在庫がある場合

[send2.2.1:]出荷を行う

[send2.2.2:]在庫を更新する

[send2.2.3:]出荷伝票を発行する

[send2.2.3.1:]注文番号

[send2.2.3.2:]客の名前

[send2.2.3.3:]酒の銘柄

[send2.2.3.4:]本数

[send2.2.4:]出荷伝票の控えを保存する

[send2.3:]在庫のない場合

[send2.3.1:]「在庫不足」を表示し謝罪する

3.2 酒を入荷

[arrive1:]卸売業者からの入荷依頼が届く

[arrive1.1:]酒の銘柄

[arrive1.2:]本数

[arrive2:]同じ銘柄の酒がある場合

[arrive2.1:]該当する銘柄の在庫本数に加算する

[arrive3:]ない場合

[arrive3.1:]新たに銘柄と本数を在庫に追加する

3.3 在庫表示

[display_stock1:]販売員からの依頼を受ける

[display_stock2:]在庫内にある全ての酒の銘柄とその本数をリストにして表示する

3.4 出荷実績の表示

[display_shipment1:]販売員からの依頼を受ける

[display_shipment1.1:]顧客名

[display_shipment2:]指定の顧客に過去に出荷した全ての出荷伝票のリストを表示する

3.5 システムを起動

[start_system:]ファイルからデータを読み込む

3.6 システムを終了

[finish_system:]ファイルに在庫状況と出荷伝票を保存する

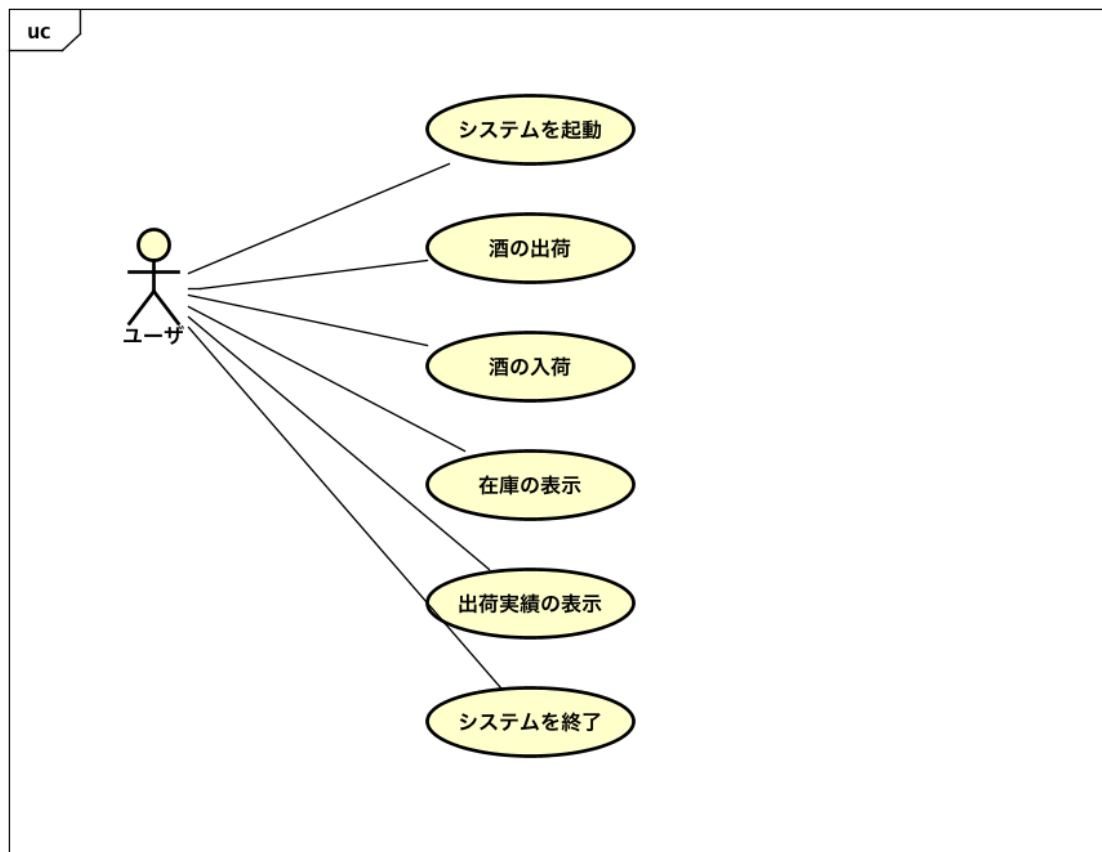
4. 非機能要求

[LANGUAGE:]C または Java で実装すること

[PLATFORM:]UNIX システム上で稼働すること

2. 分析ドキュメント

ユースケース図



ユースケース図ドキュメント

システムの起動

概要

- システムを起動し、データを読み込む

事前条件

- システムが起動されていない

イベントフロー

1. ユーザがシステムを起動する
2. システムがファイルに保存されたデータを読み込む
3. システムがメインメニュー画面(業務の選択画面)を表示する

事後条件

- メインメニュー画面が表示されている

酒の出荷

概要

- 酒を出荷する

事前条件

- システムが起動済みで、メインメニュー画面が表示されている

メインイベントフロー

1. ユーザが「出荷する」を選択する
2. システムが「客の名前、酒の銘柄、本数」を入力する画面を表示する
3. ユーザが注文内容を入力する
4. システムが、注文内容を読み込み、注文に一意的な注文番号をつける
5. システムが、指定された銘柄が存在し、かつ在庫が注文の本数より多いかを確認する
在庫がない場合 -> サブイベントフロー A へ
6. システムが、在庫を更新する
7. システムが、「注文番号、客の名前、酒の銘柄、本数」を書いた出荷伝票を作成する
8. システムが、出荷伝票を顧客に発行する

サブイベントフロー A

A.1 在庫が無い旨を顧客に通知する

A.2 メインメニューに戻る

事後条件

- 画面にメインメニューまたは出荷伝票が表示されている

シナリオ

1. 酒屋 A が「酒の出荷」を選択し、入力画面が表示される。
2. 酒屋 A が「大田和樹、ブラックニッカ、1」と入力する
3. システムが「234214」と注文番号をつける
4. システムが、在庫の有無を確認し、在庫がある場合は「214214,大田和樹、ブラックニッカ、1」と書いた出荷伝票を表示する

酒の入荷

概要

- 酒を入荷する

事前条件

- システムが起動済みで、メインメニュー画面が表示されている

イベントフロー

1. ユーザが「入荷する」を選択する
2. システムが「酒の銘柄、本数」を入力する画面を表示する
3. ユーザが、入力を行う
4. システムが、入力を読み込む
5. システムが、在庫内に同じ銘柄の酒があるかを確認する
ある場合 -> 6.A
ない場合 -> 6.B
- 6.A システムが、該当する銘柄の在庫本数を、入荷依頼の本数分加算する
- 6.B システムが、銘柄とその本数を在庫に追加する
7. メインメニューに戻る

事後条件

画面にメインメニューが表示されている

シナリオ

1. 酒屋 A が「酒の入荷」を選択し、入力画面が表示される
2. 酒屋 A が「いいちこ、12」と入力する
3. システムが、在庫に酒を追加する

在庫の表示

概要

- 在庫表示を行う

事前条件

- システムが起動済みで、メインメニュー画面が表示されている

イベントフロー

1. ユーザは「在庫表示を行う」を選択する
2. システムは在庫中の全ての酒の銘柄とその本数をリストにして表示する

事後条件

- 画面に在庫のリストが表示されている

シナリオ

1. 酒屋 A が「在庫の表示」を選択する
2. システムが「銘柄、本数」をリストにして表示する

出荷実績の表示

概要

- 出荷実績の表示を行う

事前条件

- システムが起動済みで、メインメニュー画面が表示されている

イベントフロー

1. ユーザは「出荷実績の表示を行う」を選択する
2. システムは「顧客名」を入力する画面を表示する
3. 販売員は「顧客名」を入力する
4. システムは入力された顧客名に紐づく全ての出荷伝票のリストを表示する

事後条件

- 画面に出荷伝票のリストが表示されている。

シナリオ

1. 酒屋 A が「出荷実績の表示」を選択し、入力画面が表示される
2. 酒屋 A が「大田和樹」と入力する
3. システムが、「大田和樹」に出荷した出荷伝票のすべてのリストを表示する

システムの終了

概要

- データをファイルに書き込みシステムを終了する

事前条件

- システムが起動済みで、メインメニュー画面が表示されている

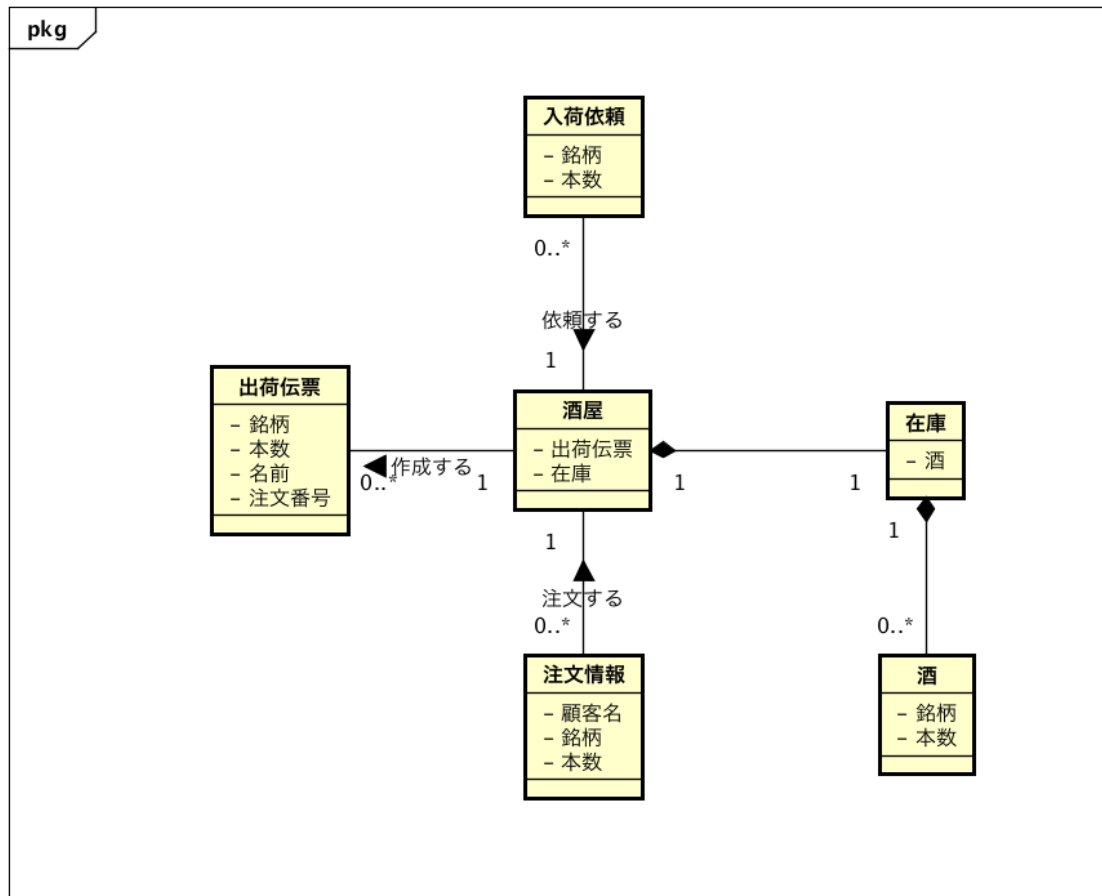
イベントフロー

1. ユーザは「システム終了」を選択する
2. システムは終了時点のデータ(在庫状況と出荷伝票)をファイルに書き込む
3. システムを終了する

事後条件

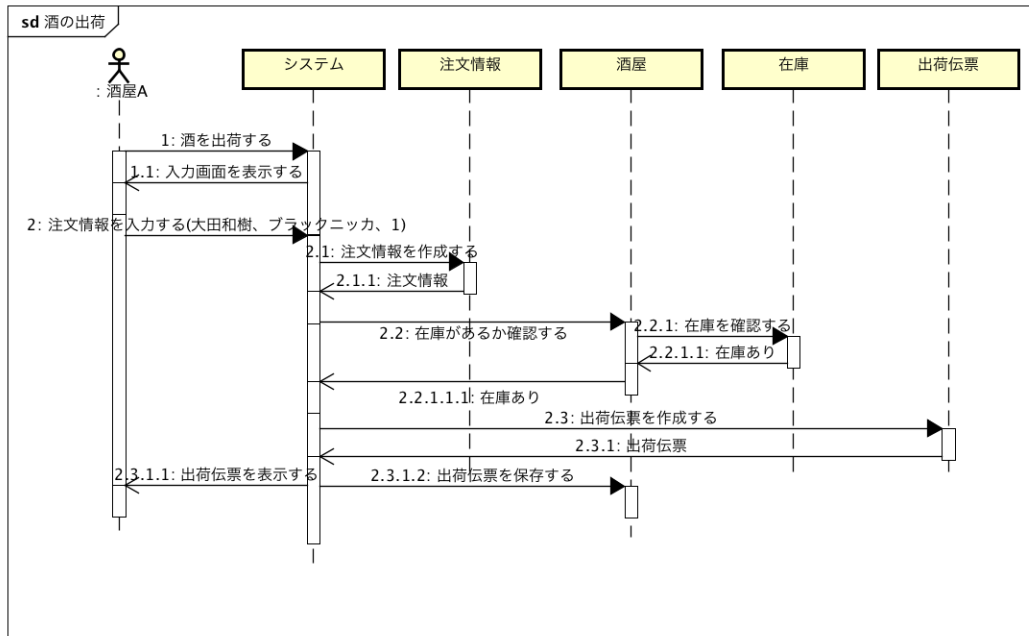
- システムが終了している

クラス図



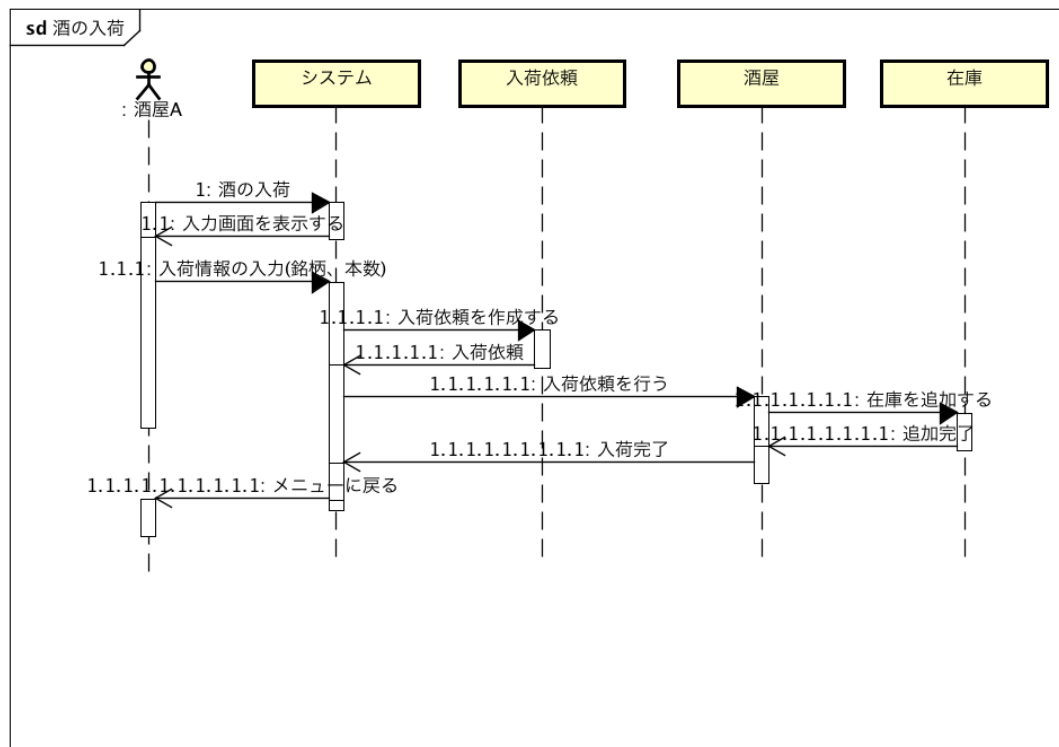
シーケンス図(システムの起動・終了のシーケンス図は単純であるため省略)

酒の出荷



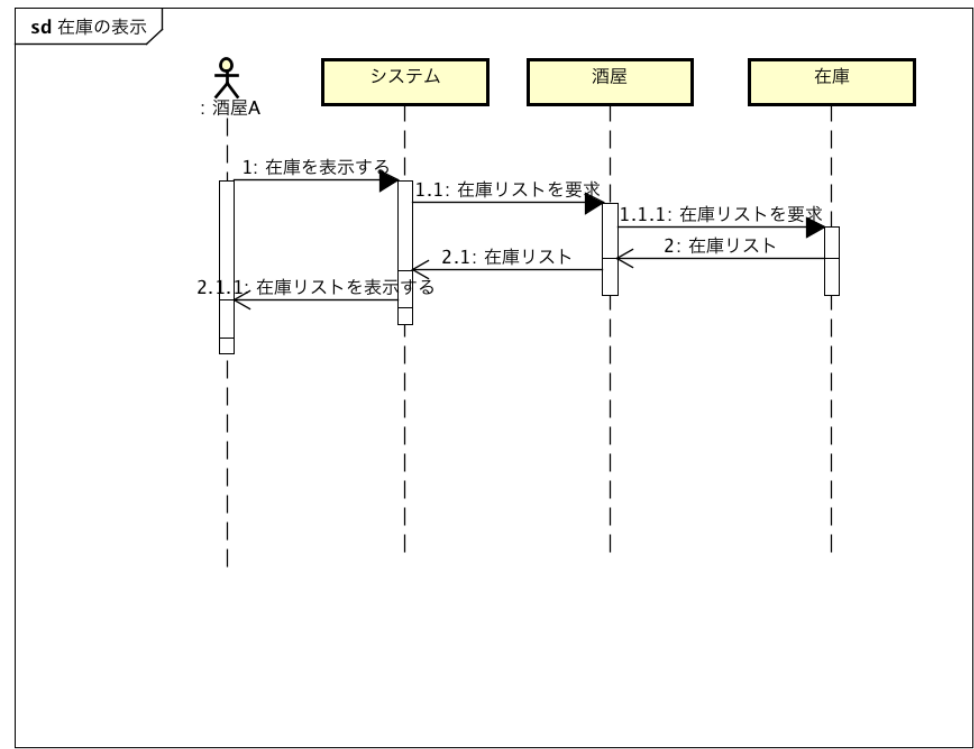
powered by Astah

酒の入荷



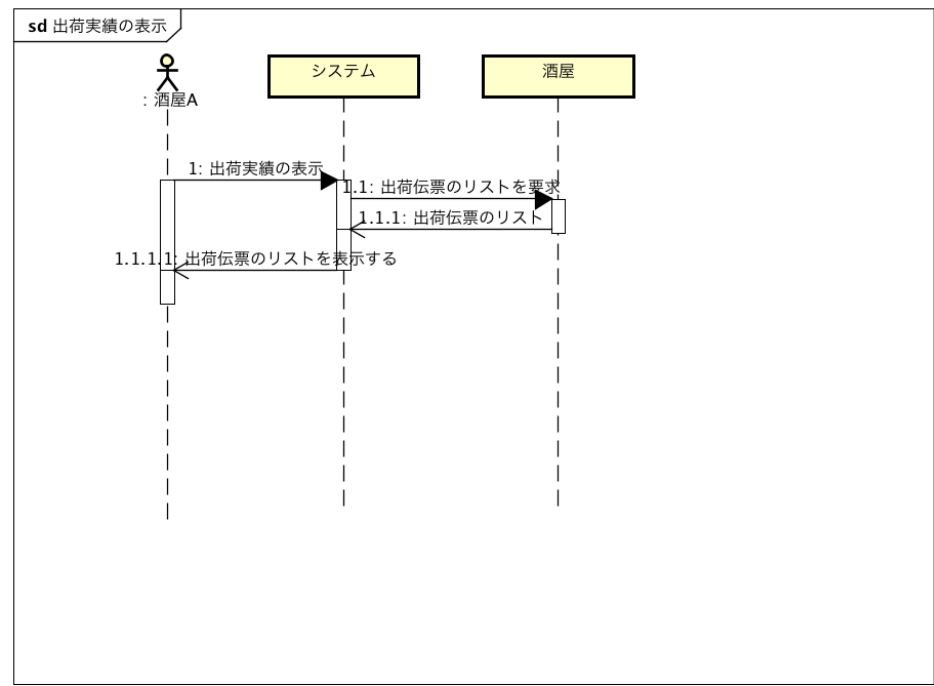
powered by Astah

在庫の表示



powered by Astah

出荷実績の表示



powered by Astah

3. 外部設計仕様書

1 はじめに

本書は酒屋在庫管理システムの外部設計仕様を定義するものである
上記の 1. 要求定義書・2. 分析ドキュメントを本書の参照とする

2 外部設計

2.1 概略

酒屋在庫管理システムはターミナル上のコンソールアプリとして表現する。

実行形式は liquorSystem という名前である。

起動方法はターミナルから以下のように入力する。

`$/liquorSystem`

起動後、ユーザは行いたい業務を選択し、システムはそれを実行する。

以下にシステムの画面イメージを示す。ユーザが、顧客「大田」からの注文をうけ、「ブラックニッカ」を「5本」出荷している場面を想定している。
(ユーザ入力部を斜体になっている)

~酒屋在庫管理システム~

行いたい業務いずれかの数字を選択して下さい

システムを終了する場合は 0 を選択してください

0: システムを終了する

1: 出荷する

2: 入荷する

3: 在庫表示を行う

4: 出荷実績の表示を行う

1

顧客の名前を入力して下さい: 大田

酒銘柄を入力して下さい: ブラックニッカ

本数を入力して下さい: 5

出荷が完了しました

~出荷伝票~

注文番号 : 000000000

名前 : 大田

酒銘柄 : ブラックニッカ

本数 : 5 本

クリックでメインメニューに戻る

2.2 要求ごとの特記事項

2.2.1 [send:]酒の出荷部

[send2:]の注文番号は一意であるものとする。

酒の在庫がない場合、その旨を表示する。

2.2.2 [arrive:]酒の入荷部

入荷した酒が在庫にある場合本数を加算し、ない場合、新たに銘柄と本数を在庫に追加する。

2.2.3 [display_stock:]在庫表示部

すべての在庫を表示する。

2.2.4 [display_shipment:]出荷実績表示部

顧客名が存在しない場合、再入力を促す。

2.2.5 [start_system:]システム起動部

システム起動時、在庫状況・出荷伝票を読み込む。

2.2.6 [finish_system:]システム終了部

システム終了時、在庫状況・出荷伝票をファイルで更新する。

2.3 サブシステムについて

システムが単純であるためサブシステムは存在しない

3 画面と入出力

3.1 画面上の書式

[[メインメニュー画面]]

はじめに、酒屋が行う業務を選択させる。

行いたい業務いずれかの数字を選択して下さい

システムを終了する場合は 0 を選択してください

0: システムを終了する

1: 出荷する

2: 入荷する

- 3：在庫表示を行う
- 4：出荷実績の表示を行う

[[入力画面 1]]

出荷業務の入力画面。

顧客の名前を入力して下さい：

酒銘柄を入力して下さい：

本数を入力して下さい：

[[入力画面 2]]

入荷業務の入力画面。

酒銘柄を入力して下さい：

本数を入力して下さい：

[[入力画面 3]]

出荷実績表示業務の入力画面。

顧客名を入力して下さい：

以下、出力画面において、X は文字、9 は数字を表している。

[[出力画面 1]]

出荷伝票の発行。

~出荷伝票~

注文番号：999999

名前：XXXXXXXXXXXXXXXXXXXXX

酒銘柄：XXXXXXXXXXXXXXXXXXXXX

本数：9,999 本

クリックでメインメニューに戻る

[[出力画面 2]]

在庫表示。

~在庫一覧~

酒銘柄	本数(本)
XXXXXXXXXXXXXXXXXXXXX	9,999
XXXXXXXXXXXXXXXXXXXXX	9,999
XXXXXXXXXXXXXXXXXXXXX	9,999

[クリックでメインメニューに戻る](#)

[[出力画面 3]]

出荷実績の表示。

~XXXXXXXXXXXXXXXXXXXXX への出荷実績一覧~

出荷伝票 1

注文番号 : 999999

名前 : XXXXXXXXXXXXXXXXXXXXX

酒銘柄 : XXXXXXXXXXXXXXXXXXXXX

本数 : 9,999 本

出荷伝票 2

注文番号 : 999999

名前 : XXXXXXXXXXXXXXXXXXXXX

酒銘柄 : XXXXXXXXXXXXXXXXXXXXX

本数 : 9,999 本

出荷伝票 3

注文番号 : 999999

名前 : XXXXXXXXXXXXXXXXXXXXX

酒銘柄 : XXXXXXXXXXXXXXXXXXXXX

本数 : 9,999 本

[クリックでメインメニューに戻る](#)

[[出力画面 4]]

出荷の際在庫がなかった場合の表示。

在庫がありません。申し訳ございません。

3.2 画面遷移

3.1 の各画面は以下の遷移関係に従い出力される。

S0: プログラムスタート

S1 へ

S1: [[メインメニュー画面]]の表示

入力値が 0 の時 S2 へ

入力値が 1 の時 S3 へ

入力値が 2 の時 S4 へ

入力値が 3 の時 S5 へ

入力値が 4 の時 S6 へ

S2: プログラムの終了

S3: [[入力画面 1]]の表示

在庫がある場合 S7 へ

在庫がない場合 S8 へ

S4: [[入力画面 2]]の表示

S1 へ

S5: [[出力画面 2]]の表示

S1 へ

S6: [[入力画面 3]]の表示

S9 へ

S7: [[出力画面 1]]の表示

S1 へ

S8: [[出力画面 4]]の表示

S1 へ

S9: [[出力画面 3]]の表示

S1 へ

4 コード設計

4.1 コード対象

- 注文番号
一意であるものとする

4.2 コード一覧

一覧が必要なコードは存在しない

4.3 コード管理

管理が必要なコードは存在しない

5 論理データ設計

5.1 データ項目

- 在庫状況
- 出荷伝票

5.2 データ関連

これらのデータ間に関連はない。

5.3 ファイル・データベース候補

在庫状況のファイルと出荷伝票のファイルを保持する。

このファイルに対し、システム起動時に読み込み、システム終了時に書き込む。