Use Case UC1: Terminplaner verwalten

Primary Actor:

Mosterei-Inhaber

Stakeholder and Interests:

* Mosterei-Inhaber: Will Termine einfacher und schneller ausmachen und verwalten können.
* Mitarbeiter: Will planen können, wie viel Kunden an einem Tag kommen.
* Kunden: Möchten einfacher und schneller einen Termin bekommen.

Preconditions:

Der Mosterei-Inhaber (Admin) hat als einziger die Berechtigung, Terminpläne zu erstellen. Hierzu gibt es gültige Anmeldedaten. Er ist im System mit den berechtigten Daten angemeldet; Daten von Kundenverwaltung

Postconditions:

Übersichtlicher Terminplan. Jeder der Mitarbeiter hat Zugriff auf den Terminplan, um für den kommenden Tag planen zu können.

Main Success Scenario:

1. Mosterei-Inhaber öffnet Terminplanung.

2. System zeigt Terminplaner an.

3. Mosterei-Inhaber gibt gewünschten Termin des Kunden ein.

4. System gibt zurück, das Termin noch frei.

5. System fragt, ob Termin passt.

6. Mosterei-Inhaber bestätigt.

7. Mosterei-Inhaber speichert den Termin ab.

8. System speichert und gibt den Terminplan an die anderen Mitarbeitern weiter, welche den Terminplan mit deren Account einsehen können.

Extensions:

\*a. Jederzeit, wenn das System ausfällt:

Um Wiederherstellung und korrekte Schichtpläne zu gewährleisten, sicherstellen,

dass alle transaktionsrelevanten Zustände und Ereignisse in jedem Schritt des

Szenarios wiederhergestellt werden können.

1. Mosterei-Inhaber startet System neu, meldet sich an und fordert

Wiederherstellung eines früheren Zustandes an.

2. System rekonstruiert früheren Zustand.

2a. System entdeckt Anomalien, die Wiederherstellung verhindern:

1. System signalisiert Fehler zum Mosterei-Inhaber, speichert den

Fehler und geht in den sauberen Zustand.

2. Mosterei-Inhaber startet Programm neu.

\*b. Der Termin muss aus verschiedenen Gründen geändert oder gelöscht werden:

1. Mosterei-Inhaber ändert den Termin oder löscht ihn.

2. System speichert den neuen Terminplan.

3. System sendet Nachricht an Mitarbeitercomputer, dass der Terminplan erneuert

wurde.

4a. Termin nicht mehr frei:

1. Kunde wählt neuen Termin aus.

2. Mosterei-Inhaber wählt gewünschten Termin aus.

3. System gibt zurück das Termin noch frei.

4. System fragt, ob Termin passt.

5. Mosterei-Inhaber bestätigt.

6. Mosterei-Inhaber speichert den Termin ab.

7. System speichert und gibt den Terminplan an die anderen Mitarbeitern weiter,

welche den Terminplan mit deren Account einsehen können.

5a. Mosterei-Inhaber bestätigt nicht:

1. Mosterei-Inhaber erstellt neuen Schichtplan bzw. bearbeitet den alten.

2. System fragt, ob Schichtplan so in Ordnung.

3. Mosterei-Inhaber bestätigt.

4. System speichert und gibt den Schichtplan an die anderen Miarbeitern weiter,

welche den Schichtplan mit deren Account einsehen können.

8a. Fehler beim Speichern

1. Mosterei-Inhaber sucht nach Fehler.

Special Requirements:

* Internationalisierung der Sprache (Englisch, Deutsch)
* Lesen von ca. 1 Meter Entfernung soll möglich sein
* In 90% der Fälle soll das Programm keine lange Ladedauer haben
* Telefon mit Headset-Funktion

Technology and Data Variations List:

Schnittstelle mit anderen Computer, Schnittstelle zu einem Drucker

Frequency of Occurrence:

Mehrmals am Tag

Miscellaneous: