**Project:** Mensa@Unibe

**Customer:** Bledar Aga

**Developers:** Raul Bolanos, Manuela Häfliger, Nicolas Kessler, Theodor Truffer

**Software Requirement Specification**

1. **Introduction**

The application gives students, employees or any interested person informations about the mensas of the University of Berne. Among these informations should be the locations of all the mensas, how to get there from the current position and of course the daily menu and the ones of the following days with all its prices. The app will include basic social media functionality, such as inviting friends for lunch or rating the menus.

The application will be running on Android based devices.

The goal of this application is to help students find the right mensa to have lunch.

1. **Use Cases**

(index by importance)

#1

Name: List

Description: Get a list of all mensas on the campus.

Basic Flow: 1. The user starts the application

2. The user taps on “List” in the action bar and the application shows the list of all available mensas

#2

Name: Details

Description: Get detailed information about a certain mensa such as schedule, daily menu, menus of the next days, prices, employees, address.

Basic Flow: 1. The user starts the application

2. The user taps on “List” in the action bar and the application shows the list of all available mensas

3. The user taps on the mensa he wants information about and the application shows all the details about that mensa

#3

Name: Closest mensa

Description: Find and show the closest mensa from your current position .

Basic Flow: 1. The user starts the application

2. The user taps on “Closest mensa” in the action bar and the application shows with a mark the closes mensa from the users position on a map

#4

Name: Find mensa

Description: Shows the way on the map to a chosen mensa.

Basic Flow: 1. The user starts the application

2. The user taps on “List” in the action bar and the application shows the list of all available mensas

3. The user chooses the mensa he wants to go and the application shows information about this mensa

4. The user clicks a button to find this mensa and the application shows the map and the shortest way to this mensa from the users current position

#5

Name: Favourite mensa

Description: Choose a favourite mensa, for which you can quickly see the current menu and the menus for the next days.

Basic Flow: 1. The user starts the application

2. The user taps on “Settings”

3. The user taps on “Favourite mensa”

4. The user can choose his favourite mensa from a list of available mensas

5. Once this preference is set, the user gets all important informations on the start-screen after launching the application

#6

Name: Add Friends

Description: Add other users of this application to your friendlist.

Basic Flow: 1. The user starts the application

2. The user taps on “Friends” in the action bar and the application shows a list of all added friends

3. The user taps on the button “add friend”, enters a friend’s email adress to add him to the list

#7

Name: Notify Friends

Description: Let all your friends, or specific ones, know where you go for lunch, so they can join you.

Basic Flow: 1. The user starts the application

2. The user taps on “Friends” in the action bar and the application shows a list of all added friends

3. The user taps on the button “invite”, chooses all or individual friends and the mensa he plans to go to

4. the user gets a notification as soon as his friend joins or declines the event

#8

Name: Join friends for lunch

Description: If a friend lets you know where he goes for lunch, you can join him.

Basic Flow: 1. The user gets an invitation by push notification

2. The user starts the application

3. Above the daily menu of the user’s favourite mensa appears the friend’s invitation with two buttons “join” or “decline”

4. The application sends a notification with the answer back to that friend

#9

Name: Rate a menu

Description: Grade a menu you’ve eaten from zero to five

Basic Flow: 1. The user starts the application

2. Either the user taps on a menu of his favourite mensa on the start-screen or taps on “List”, taps on the mensa where he had lunch or taps on “Upcoming Menus”, chooses the menu he had

3. The user taps on “Rate” and can now rate the menu from zero to five

#10

Name: Menu’s rating

Description: See how other people rated a specific menu

Basic Flow: 1. The user starts the application

2. Either the user taps on a menu of his favourite mensa on the start-screen or taps on “List”, taps on the mensa where he had lunch or taps on “Upcoming Menus”, chooses the menu he had

3. The user can see next to the menu how many people have rated this menu and the average grade

#11

Name: Favourite food

Description: Get notified when your favourite mensa serves your favourite food

Basic Flow: 1. The user starts the application

2. The user taps on “Settings”

3. The user taps on “Food criterias”

4. The user can choose from a list of ingredients his prefered ones

5. Once this preferences are set, the user gets a push notification when his favourite mensa serves a menu containing at least one of the user’s favourite ingredients

#12

Name: Upcoming menus

Description: Lets the user browse all upcoming menus of the week, including the price of the menus and where they will be served.

Basic Flow: 1. The user starts the application

2. The user taps on “Upcoming Menus” in the action bar

3. A list will be displayed with all menus of the week, sorted by the day when they will be served.

#13

Name: English menus

Description: Shows the english translation of a menu

Precondition: The menu is not already in English

Basic Flow: 1. The user starts the application

2. Either the user taps on a menu of his favourite mensa on the start-screen or taps on “List”, taps on the mensa where he had lunch or taps on “Upcoming Menus”, chooses the menu he had

3. The user taps on an icon displaying the union jack

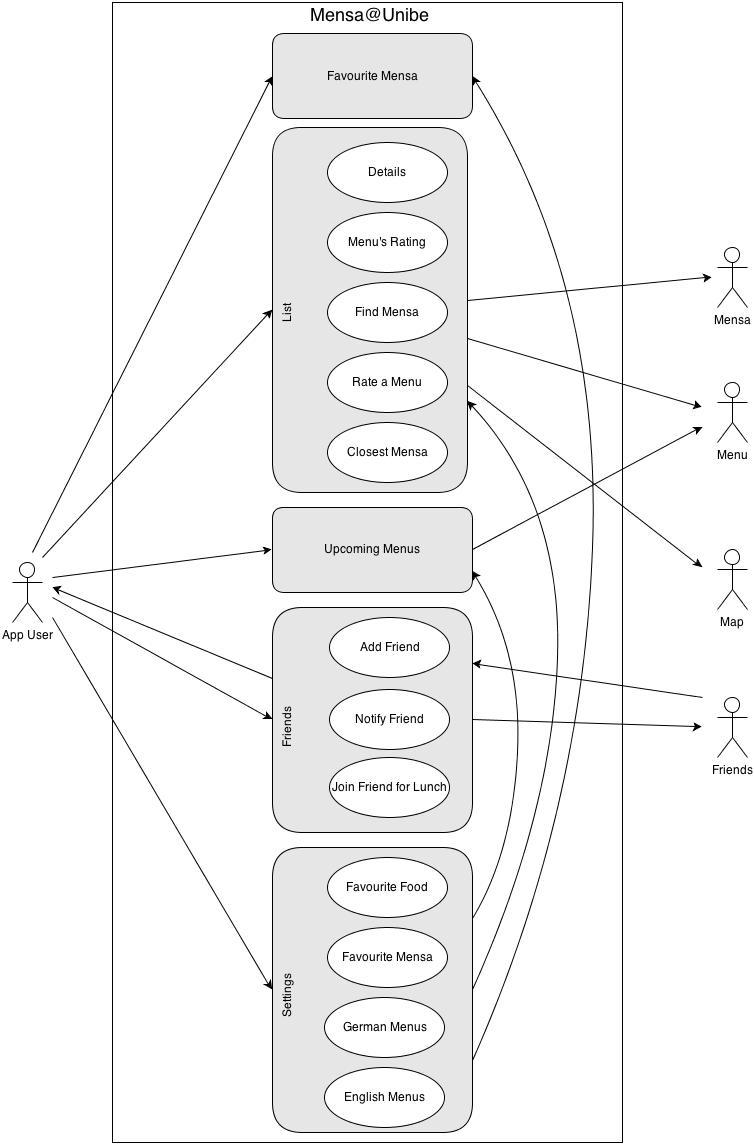
#14

Name: German menus

Precondition: The menu is not already in German

Description: Switches an english menu back to german

Basic Flow: The same as in use case #13 but instead of a union jack in step 3 there will be a german flag.



1. **Specific requirements**