# Project Requirements

## Project Name:  MensaUniBe

## Team: 6

## Customer:  Bledar Aga

Revision History

| Version | Date | Revision Description |
|---|---|---|
| .01 | 2.10.13 | Initial version of the SRS document |
|  |  |  |
|  |  |  |
|  |  |  |

Date:   October 2, 2013

# Introduction

## Purpose

The purpose of this app is to provide various informations about the different mensas of the university of Bern. These informations help the user to decide where to go for lunch, find any mensa on the campus or share his mensa experience with his friends by inviting them for lunch and/or rating menus.

## Stakeholders

### Users

Students and university employees who are hungry and have an Android device. The users need to have an app which allows them to quickly find out everything they need to know about the mensas and the served menus.

### Client

Our customer Bledar Aga represents the mensa company. Bledar needs a working app which is delivered within the time schedule.

### Developers

Highly motivated but not very well paid. They are eager to create a quality app.

## System Overview

MensaUniBe Android App:

This is the interface for the users to the whole system. This app interacts with the Mensa-Webservice and our Social-Webservice.

Mensa-Webservice:

Webservice which provides all informations about the mensas and their daily or weekly menu plans. (https://github.com/lexruee/Mensa-Webservice)

Social-Webservice:

Handles the menu ratings and interactions between different users. Is separated from the external Mensa-Webservice.

# Overall Description

## Use Cases

### Overview

**0. Browse favorite mensas**

Our average student or Employee of the University of Bern has enough work to do besides looking for the best place to eat something as trivial as lunch. So to simplify this, he or she has the possibility to see all his or hers  favorite mensas and the food they serve in just a glimpse at the moment they start the application.

**1. Add mensas to favorites**

In order to show all favorite Mensas (See Use Case 0.) the user  acctually need to have a favorite mensa and also needs to add them to the System. So if the user wants to do this he can shost press a button and voilà the mensa is now a favorite. Pressing the same button again will remove the mensa from the list of favorites.

**2. Browse all mensas**

The user needs to get an overview of all mensas and their menus in order to select his favorites, find directions to any of them or just to look for decent food.

**3. Browse all menus of the current day**

The main meaning of the app is to have a daily or weekly overview of all the menus that are served in the mensas around the university of Berne. The app provides multiple ways to get such overviews. The users can either have a list of all menus of one mensa or they can see a list of all menus from the mensas they have chosen as favourites.

**4. Find way to the mensas**

The user wants to find the way to any mensa on the campus, so he is abel to do so by just clicking the right button from the chosen mensa and gets guided.

**5. Invite friends for lunch**

The user does not like to eat alone, so he decides to invite some friends for lunch. He starts the MUB-App and clicks on the Social Button/Tab which leads him to the screen where all active invitations are listed (i.e. invitations which he received or sent recently). He doesn’t bother to look at these, he just presses the Send-New-Invitation button. This opens the dialog where he selects the mensa where he wants to eat at and also enters a time and a date. Then he adds some of his friends to the receivers list and presses the “Send” button which then sends the invitation to his friends if an internet connection is available.

**6. Reply to invitation**

If the user didn’t disable notifications in the MUB-App settings, he will be notified if anyone invites him automatically (via a android notification). The notification allows the user to directly get to the part of the app where all invitations are listed. If notifications are disabled, he just can open the app and check manually if there are any new invites.

As the user sees that he got a new invitation, he navigates to the screen where all invitations are displayed with all their details. There he can also directly accept or decline an invitation.

### 7. Check invitation status

The user sent an invitation to his friends some time ago, so he wonders if they come or not. In the invitations list he sees all of his sent invitations (below the received invitations). He clicks on the invitation of interest and sees the details of the invitation and also a list of persons he invited. Any person who accepted the invitation is marked with a green checkmark, friends who didn't want to eat with the user are marked with a red cross. Anyone who didn't answer yet just has a questionmark. If the user decides he wants to see more of his friends for lunch, he just presses the "Invite more" button and selects other friends which should receive the invitation from the friends list.

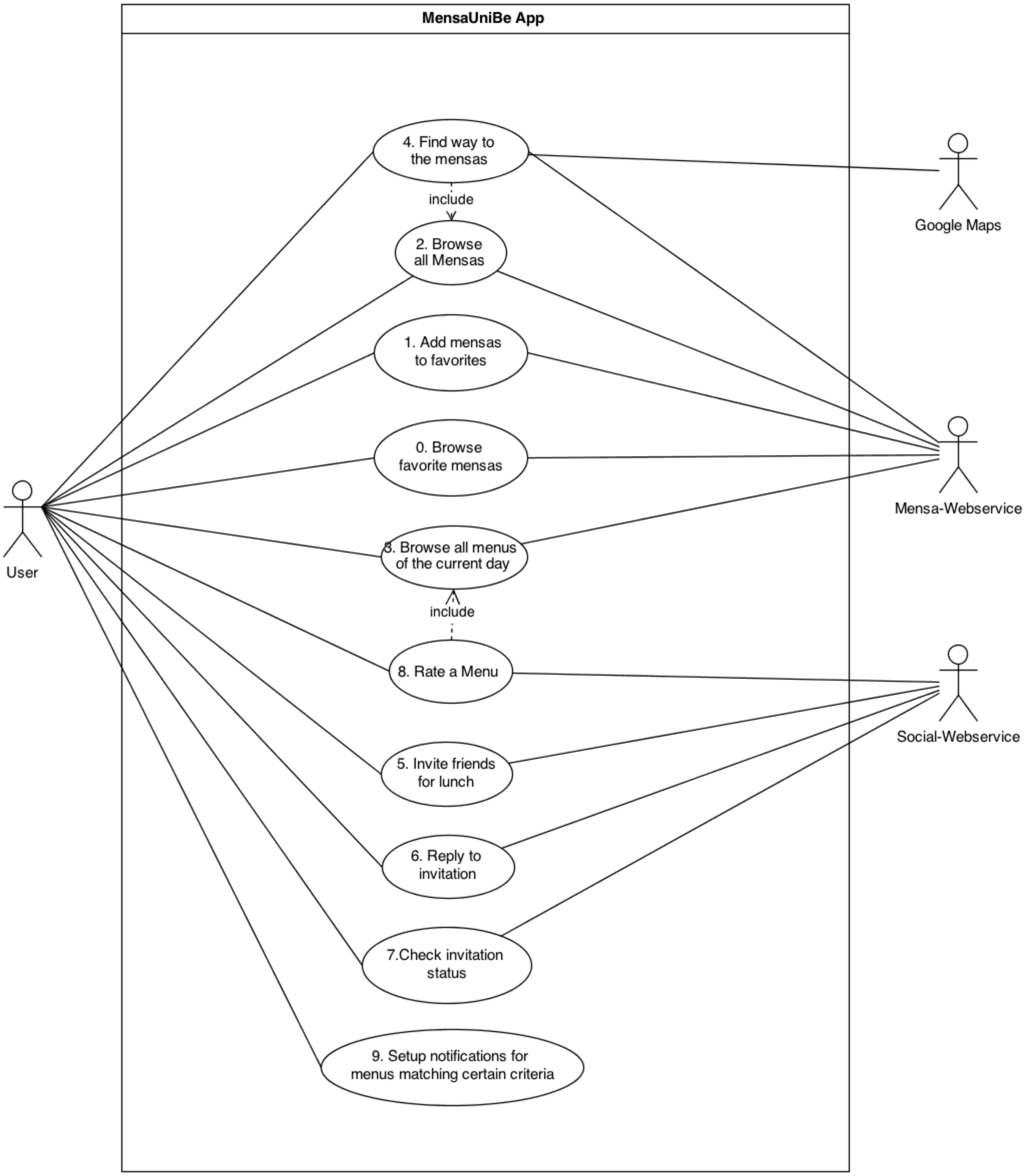
### 8. Rate a menu

Users want to share their experiences with others. That's why they have the ability to rate the menus in the app. It's just one click on the menu and a small feedback how the menu tasted. Anyone we'll be able to see the user's rating, which supports the user in making the decision where to eat.

### 9. Setup notifications for menus matching certain criteria

Some users don't want to check the menus every single day, because they are might be a bit choosy of what they eat and they find only seldom something they like. So the app lets the user select criteria which he wants to be notified of the day its served. This is achieved in selecting a set of criteria and a set of mensas in the apps SetUp Notification Option.

## Diagram


MensaUniBe App
4. Find way to the mensas
include
2. Browse all Mensas
1. Add mensas to favorites
0. Browse favorite mensas
3. Browse all menus of the current day
include
8. Rate a Menu
5. Invite friends for lunch
6. Reply to invitation
7.Check invitation status
9. Setup notifications for menus matching certain criteria
User
Google Maps
Mensa-Webservice
Social-Webservice

# Detailed Use Cases

## 0. Browse favorite mensas

1. **Actors**

   User (primary)
   Mensa-Webservice (secondary)

2. **Description**

   The average Student of the University of Bern wants to see his favorite Mensas and is able to see their menus in just a glimpse.

3. **Trigger**

   The user starts the application by pressing the application's Icon.
   The application starts and automatically shows all the Mensas which are previously marked as favorite Mensas
   If there are no Mensas marked as favorites, all Mensas are shown instead or a default logo will be displayed

4. **Preconditions**
   1. For optimal use favorite Mensas must have been added.
   2. If it is the first use of the application a welcome is displayed.

5. **Postconditions**
   1. Student is now informed about the food served in his favorite Mensas.

6. **Main Scenario**
   1. User presses the icon and starts the app
   2. App retrieves mensa and menu information from the Mensa-Webservice
   3. App shows the favourite mensas the food they serve
   4. User has the information he needs and closes the app

7. **Alternative Scenarios**

   4a. The User has no favorite Mensas yet
   1. All Mensas are displayed
   2. User can add a Mensa to his favorites by pressing the star button next to the Mensa name

   4b. App is started for the first time
   1. A Welcome note is displayed followed by Instructions

8. **Notes**
   1. The favorite Mensa screen is displayed as Home view, should it be accessible through other menu options?
   2. What if a user doesn't have a favorite mensa and doesn't want to add one => all mensas at system start if no favorites are found?

## 1. Add Mensa to Favorites

1. **Actors**

   User (primary)
   Mensa-Webservice (secondary)

### 2. Description

If you are in the Mensa Browser or have selected a Mensa it is possible to add a Mensa to your favorites.

### 3. Trigger

The user starts the application by pressing the application's Icon.
The application starts and automatically shows all the Mensas which are previously marked as favorite Mensas
If there are no Mensas marked as favorites, all Mensas are shown instead or a default logo will be displayed.

### 4. Preconditions

1. The User is in the Mensa Browser or has selected a Mensa.

### 5. Postconditions

1. From now on the Mensa is shown as favorite Mensa and when then app is started.

### 6. Main Scenario

5. User starts app for the first time
6. He or she searches in the Mensa list for a Mensa
7. The Mensa which is the users most visited Mensa is selected
8. The User presses the Button for marking as favorite
9. User can now find this Mensa under favorites

### 7. Alternative Scenarios

4a. Adding Mensa directly from the Mensa Browser

1. All mensas are displayed
2. User press the mark button

### 8. Notes

1. To remove the app you just have to press the Button again
2. The User can also favorite more than one mensa.

## 2. Browse all mensas

### 1. Actors

User (primary)
Mensa-Webservice (secondary)

### 2. Description

As a user I want to see what to browse the mensas and their menues sorted by different criteria.

### 3. Trigger

User starts the app and...
a) ...navigates to "Browse Mensas".
b) ...selects sorting criteria.

### 4. Preconditions

1. User has a connection to the internet or has already downloaded the menus of the whole week.

### 5. Postconditions

1. All menus of the selection are listed on the screen.

### 6. Main Scenario

1. User opens the MUB App
2. User navigates to “Browse menus”
3. App updates the content from the Mensa-Webservice if it has not already downloaded.
4. User defines how he want the content to be displayed.

### 7. Alternative Scenarios

1. If there's no connection to the Mensa-Webservice: Message on the screen “Cannot download mensa data, please try again later”

### 8. Notes

## 3. Browse all menus of the current day

### 1. Actors

User (primary)
Mensa-Webservice (secondary)

### 2. Description

a) As a user I want to see what menus are served at the current day.
b) As a user I want to see what menus from the list of my favourite mensas are served at the current day

### 3. Trigger

User starts the app and...
a) ...takes a look at the list of menus of today
b) ...navigates to his “Favourites”

### 4. Preconditions

1. User has a connection to the internet or has already downloaded the menus of the whole week.

### 5. Postconditions

1. All menus of the current day are listed on the screen
2. All menus of the current day from the list of my favourite mensas are listed on the screen.

### 6. Main Scenario

a)
1. User opens the MUB App
2. User goes to where the page with the listed menus is.
3. App updates the content from the Mensa-Webservice if it has not aleady downloaded.
4. User sees the list of today’s menus.

b)
1. User opens the MUB App
2. User goes to the section of the “Favourites”
3. App updates the content from the Mensa-Webservice if it has not aleady downloaded.
4. User sees the list of today’s menus (only those of his favourite mensas).

### 7. Alternative Scenarios

Both in a) and b):

3a. App cannot connect to Mensa-Webservice

1. App shows an error: “Cannot download mensa data, please try again later.”
2. Use case ends

### 8. Notes

## 4. Find way to the mensas

### 1. Actors

User (primary)
Mensa-Webservice (secondary)
Google-Maps (secondary)
GPS-Module (secondary)

### 2. Description

a) As a user I want to know where the mensas are.
b) As a user I want to get directions to the different mensas.

### 3. Trigger

User starts the app and...
a) ...navigates to the “Navigation”-Tab.
b) ...clicks on the “Get directions”-Button next to a specific mensa.

### 4. Preconditions

1. User has a connection to the internet.
2. Connection to google maps is possible.

### 5. Postconditions

a. All mensas are displayed in a map.
b. The directions to the selected mensa is displayed on a map.

### 6. Main Scenario

1. User opens the MUB App
2. Trigger a) or b) applies:

3a. All available mensas are displayed on a map
4a. Clicking on the mensa opens an overview trigger
3b. The directions between the phone’s location and the selected mensas are displayed on a map

### 7. Alternative Scenarios

1. If there's no connection to the internet:
   Message on the screen "No connection to the internet."
2. Content not loaded:
   Message on the screen "Menues could not be downloaded."
3. Google Maps not accessible:
   Message on the screen "No connection to mapping service."

### 8. Notes

1. Google maps will probably give good results. No alternative needed.

## 5. Invite friends for lunch

### 1. Actors

User (primary)
Social-Webservice (secondary)
Mensa-Webservice (secondary, only used in alternative scenario 3a)

### 2. Description

As a user I want to invite some friends for lunch in a mensa.

### 3. Trigger

The user starts the app.

### 4. Preconditions

1. The user has at least one friend who uses the MUB-App. If not, inviting friends does not make sense.

### 5. Postconditions

1. The invitation is distributed to the addressees by the Social-Webservice.
2. The invitation gets listed as “Sent invitation” in the invitations list.

### 6. Main Scenario

1. User switches to the "Social" tab
2. User presses the “Create new invitation” button
3. App retrieves the list of available mensas from the cache if possible.
4. App switches to the “Create invitation” dialog
5. User selects mensa from the available mensas, day and time for the invitation
6. User presses the “Add receivers” button
7. App accesses the friends list and displays the list of friends
8. User selects one or more friends
9. User adds these friends as receivers by pressing the “ok” button
10. App returns to the “Create invitation” dialog and lists the selected friends as receivers
11. User presses the “Send invitation” button to send the invitation to the receivers
12. App connects to the Social-Webservice if an internet connection is available
13. App returns to the "Social" tab
14. App sends the invitation to the Social-Webservice
15. Social-Webservice sends invitation to the addressees which can then accept or decline the invitation

### 7. Alternative Scenarios

3a. List of mensas not available in the cache

1. App retrieves the list from the Mensa-Webservice
2. App stores the list in the cache
3. Use Case resumes on step 4

3b. List of mensas not available in the cache

1. App cannot receive the list
2. App shows a connection error message saying that the mensas cannot be retrieved at the moment and the user should try again later and ensure he has a working network connection.
3. Use Case is cancelled and the app returns to the "Social" tab

11a. User didn’t fill in all fields properly (either no mensa, no time, no date or no receivers specified)

1. App prompts a message which says: “Please make sure to select a mensa, a time, a date and at least one receiver”
2. User presses “ok” button
3. Use Case resumes right before step 11 and the App allows the user to fill in any missing information (or cancel the creation of the invitation by pressing the “Cancel” button)

12a. App cannot connect to the Social-Webservice

1. App shows error
2. Use case is cancelled [IMPROVEMENT NEEDED, see Note 3.]

### 8. Notes

1. The user can also add friends before selecting mensa, data and time. So step 5 could also just go after step 10
2. Should the user also be able to enter a **short custom message** as part of the invitation? Something like “hey guys, lets meet for lunch to discuss the ESE project”. This could also be added later if needed/desired.
3. Could the connection testing for the Social-Webservice be done before even opening the create invitation dialog? Maybe when the "Social" tab is opened it could prompt an error if the server is not available and grey out all options which require the Social-Webservice.

## 6. Reply to invitation

### 1. Actors

User (primary)
Social-Webservice (secondary)

### 2. Description

As a user I want to reply to a received notification.

### 3. Trigger

The user either gets a push notification from the Social-Webservice telling “You got a new invitation” or just navigates manually to the "Social" tab and sees the new invitation in the list of invitations.

### 4. Preconditions

1. User is in the "Social" tab, either by manually navigating there or by clicking on the “Show” button of the notification

### 5. Postconditions

1. The reply is sent to the sender of the invitation.
2. The invitation gets listed in the "Social" tab as “upcoming event” if accepted.

### 6. Main Scenario

1. App displays invitation at the top of the invitations list, showing all its details and also small accept and decline buttons. (only small icons, no button text needed)
2. User sees the invitation(s) he needs to reply to.
3. User either presses accept or decline
4. App sends the reply to the Social-Webservice
5. Social-Webservice sends reply to the person who invited for lunch.

### 7. Alternative Scenarios

4a. Reply cannot be sent

1. App shows an error: “Cannot send reply at the moment, please try again later”
2. Use Case ends and App just shows the "Social" tab.

### 8. Notes

1. If the reply cannot be sent, it would also be possible to add the reply to a queue and just send it as soon as possible. But this would mean the app should periodically check if it can access the Social-Webservice, and this would in turn consume too much power. So in this case it’s probably better to just notify the user and then do nothing until the user manually retries to send the reply.
2. Should the reply also trigger a notification for the one who invited? Are would that be “too much” notification spam? This could also be added in a later phase of the project if needed (low priority).
3. Should the reply actually be sent to the sender of the invitation or just entered in the database and then retrieved if the user wants to see the replies?

## 7. Check invitation status

### 1. Actors

User (primary)
Social-Webservice (secondary)

### 2. Description

As a user I want to see the replies to my invitation(s).

### 3. Trigger

The user navigates to the "Social" tab.

### 4. Preconditions

1. User has sent at least one invitation which has not passed yet.

### 5. Postconditions

3. User knows who accepted, declined or ignored the invitation.

### 6. Main Scenario

5. User clicks on a invitation he sent
6. App connects to the Social-Webservice
7. App retrieves replies from all invitees from the Social-Webservice
8. App opens a dialog which shows the details of the invitation and the replies (shows with checkmark who accepted, cross who declined and questionmark who did not reply yet)

### 7. Alternative Scenarios

2a. App cannot connect to Social-Webservice

3. App shows an error: “Cannot retrieve replies at the moment, please try again later”
4. Use Case ends and App just shows the "Social" tab.

### 8. Notes

## 8. Rate a menu

### 1. Actors

User (primary)
Social-Webservice (secondary)

2. **Description**

   As a user I want to see the ratings of a menu and be able to rate it.

3. **Trigger**

   User opens the app with the intention to rate a menu.

4. **Preconditions**

   1. User has a connection to the internet

5. **Postconditions**

   1. The rating has been sent to the Social-Webservice.
   2. The MensaUniBe app has updated the ratings so that the user is sure his rating has been uploaded.

6. **Main Scenario**

   1. User opens MensaUnibe App
   2. User navigates to the menu he wants to rate.
   3. User taps on the Rate-Button.
   4. User defines how to rate the menu and accepts.
   5. App sends the rating to the Social-Webservice.
   6. App downloads the rating-content from the Social-Webservice.
   7. The ratings from the displayed menu are being updated and the user can see his new rating.

7. **Alternative Scenarios**

   If there's no connection to the internet:
   1. Rating screen does not open.
   2. Message on the screen "Connection to the internet required."

8. **Notes**

   1. Can you only rate between a certain range of numbers or is the rating just a small text?
   2. If there's no connection to the internet, the Rate-Button could also be greyed out so that it couldn't be activated.

### 9. Setup notifications for menus matching certain criteria

1. **Actors**

   User (primary)

2. **Description**

   The User has the possibility to select criteria he wants to be notified of if a menu that matches is served that day. He can choose to be notified of all theMensas, his favorite Mensas or a single Mensa.

3. **Trigger**

   User Presses the Setup Notification Button in the Application’s Options

4. **Pre-conditions**

   If the user wants to be notified of his favorite mensas, favorites must have been chosen before. If he hasn't chosen favorites before a Message will be displayed.

## 5. Post-conditions

From this on the application sends push messages via android notification when a menu in the selected set matches a criteria.

## 6. Main Scenario

1. User goes to "Setup Notification" in the App's options
2. User selects one or more criteria from the criteria set which is displayed
3. User selects a set of mensas
4. The view closes after the "ok" button is presse

## 7. Notes

1. When the notification is clicked, the user sees the view of the mensa the menu is served.
2. Where is the Button exactly implemented in the design e.g. Tab or Menu Point?
3. When should the Notification be displayed (can the user determine this)?

# Specific requirements

## Functional requirements

- Retrieve and display menu data from the mensa webservice
- Cache menu and mensa data to allow the app to be used without internet connection.
- Get location of a specific mensa and show this mensa on the map (using a third party map API, i.e. google maps).
- Calculate directions and distance to a mensa.
- Manage the list of favorite mensas.
- Manage menu criteria and notify user if a menu matches the user's criterias.
- Translate mensa menus at least into english.
- Send message to other users (invitation or reply to invitation) using the Social-Webservice
- Social-Webservice needs to distribute user messages to the addressees.
- Retrieve messages from the Social-Webservice.
- Send a rating to the Social-Webservice and receive ratings from the Social-Webservice. The ratings need to be associated with a the menu's string representation to reuse ratings if a menu is served again.
- Manage a list of friends who also use the MUB-App. (probably via the already existing contacts)

## Non-Functional requirements

- The Mensa-Webservice and Social-Webservice should be available. If not, the app can only display already downloaded data and not manage ratings, invitations and replies.
- The app should be self-explanatory.
- It should be more comfortable to view the mensa menus via the MUB-App than the mensa homepage. If not, there is no use for the app.
- The project should be finished in time.