*Product Sheet*

*Requierments*

**MeetnLunch**

|  |  |
| --- | --- |
| Description of the document |  |
| Title | Requierments |
| Date | 05/04/2017 |
| Author(s) | *50161375* ***KAUCH Jonathan***  *50161378* ***TOMA Ivan***  *50161387* ***FRAYSSE Christophe***  *50161388* ***LIN Chaohui***  *50161390* ***MOUTIER Baptiste***  *50161389* ***ZABANDITH Brian*** |
| Subject | <Product Sheet of requirements of MeetnLunch Mobile Application> |
| Version | 1.0 |

Tableau de Révisions :

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Date | Version | Auteur | Section(s) | Commentaire |
| 05/04/2017 | 1.0 | Ivan TOMA  Jonathan KAUCH  Christophe FRAYSSE  Chaohui LIN  Baptiste MOUTIER  Brian ZABANDITH | N/A | Création du document |

Summary

Requirements

MeetnLunch

.........................................

1 **Product description**

.................................................................

2 **UI Diagram**

..................................................................

3 **Uses cases**

..................................................................

4 **Process description**

..................................................................

5 **Meeting reports**

**1**

ICT Software engineering

Product Sheet Description

Mobile Application

**MeetnLunch**



    It’s lunchtime at work but had something to finished so you could not had lunch with your co-workers and ended being alone? Or you want to go eat somewhere outside but none of your friends are available?

MeetnLunch can solve your problem! Just grab your phone and go wherever you want to eat, people will be aware through the application that you looking for someone to eat with and will join you. You can also just go outside and search for some people whose also using the app and looking for someone.

It’s a good opportunity to meet new people or even make friends!

**Account System**

The application deliver an account manager and interface for the users.

*Login Page*

* Connect with Nickname or email address
* Retrieve your password through “Forgot password ?”

*Register  Page*

* Real Name
* Nickname (mandatory for connection)
* Email (For retrieving lost passwords)
* Choose a password
* Specify your age and your gender

**Application**

The application will display a map of the neighborhood, link to the google API.

User will be able to switch between three screens via a menu at located at the bottom of the screen. This one will contain:

“Search” which is the default application page with the map displayed,

“Filters” allow the users to search for some kind of people in particularly or to be visible for some others,

“Profile” allowing the you the customize your profile

*SearchView*

* Map display
* Pop-up windows showed on the map representing people searching for some company containing a summary of the information provided by the user, click on them to see the all the details.
* Restaurants nearby recommended by the app

*Profile View*

* Picture (Avatar) - if None nothing will be displayed
* Name / Nickname - Show your nickname or your name
* Hidden Mode -  Be visible or not on the map
* Gender - Display or not your gender
* Age - Display or not your age
* Contact - Any kind of contact as phone number, kakao ID
* Description - Any additional informations to display

*Filters View*

* Range - How far should the application search for
* Kind of Food - Choose what you envy the most
* Age - Choose the range age of the people you want to meet
* Only Visible for - Apply an age and gender filter on which kind of people can see you

***Developers Team***

*50161375* ***KAUCH Jonathan***

*50161378* ***TOMA Ivan***

*50161387* ***FRAYSSE Christophe***

*50161388* ***LIN Chaohui***

*50161390* ***MOUTIER Baptiste***

*50161389* ***ZABANDITH Brian***

**2**

ICT Software engineering

Product Sheet UI DiagramC:\Users\Ivan\Downloads\ICT UI.png

**3**

ICT Software engineering

Product Sheet Use Case and Sequence Diagram

Use case: Search a lunch partner



* **Primary Actor : User**
* **Precondition :**The user has logged in
* **Main success scenarios :** 
  1. User selects a place or a food and enters the criteria of lunch partner
  2. System shows a list of suggested users
  3. User sends the invitations to users
  4. System accepts the match
  5. System sends lunch partner’s informations to the user
* **Exception scenarios :**

    1) The place doesn’t exist or the food isn’t in our database.

    2) System checks if this user is blocked or ban.

* System sends the reject message to user and the expiration date of the ban

    5) System accepts the match if the lunch partner accepts the invitation.

Use case: Sequence Diagram

C:\Users\Ivan\Downloads\sequence meetnlunch.png

**4**

ICT Software engineering

Product Sheet Process

**Software Tool Set:**

Android Studio 2.3 for Android Development

PostMan Chrome Extension for API testing

Github

**Group organization**

Baptiste Moutier API Developer

Brian Zabandith API Developer

Ivan Toma Mobile Developer

Jonathan Kauch Mobile Developer

Christophe Fraysse Mobile Developer

Chaohui Lin UI/UX Designer

**Schedule:**

Beginning of April to end of May 2017:

- Application/API development. Core features will be developed during this period (connection, registration, geolocation and eating partners searching).

Beginning of June to Final Presentation:

- Feedbacks bugs corrections, Final Project documents realization.

**Development Task List:**

- UI Design

- Login/Registration

- Forgot Password Feature

- API Connection

- Localisation

- Filters

- Profile Settings

- Show other users on map

- Detect restaurants

- Pop-up Window on map and full detail scroll

**Security Management:**

Security management is present at several points in the software development cycle.

There is a risk of attack when the software is in production. But we must also pay attention to Development process itself.

**Project Accommodation and Infrastructure:**

The development process will be realized thanks to Git and in particular Github. All repositories for MeetnLunch will be private, only members of the group can access it. The members of the group have access to different directories in the project, and have the necessary permissions to read and write on the directory, contribute to the code and use the various tools provided by Github. Git allows to have a history of changes to the code because each change is signed and authenticated.

Access is realizes via an SSH connection, which is a secure communication protocol, it imposes an encryption key exchange at the beginning of the connection, thereafter all the data exchanged are authenticated and encrypted. Code changes are replayed between members of the group in order to ensure that it does not bring vulnerabilities and does not impact other components of the project.

**Software security :**

MeetnLunch makes these exchanges with the database via our API.

The API is available via the HTTPS protocol, which should guarantee the confidentiality and data exchanged by users (forms etc.). User Authentication passes also by this API. Sensitive user data, such as passwords, is encrypted before being saved as a database, to ensure the confidentiality of the contents. The exchanges between users and server are realized via a unique "token" system for each user, defining these possible interactions with our API, in this way a user does not have access to the data of another. But also many vulnerabilities can be avoided in accordance with good development.

**5**

ICT Software engineering

Meeting Report