University for Applied Sciences Informatics Department Applied Informatics

LionsApp-Documentation

|  |  |
| --- | --- |
| Clients: | Prof. Dr. Stephan Kurpjuweit,  Jens Kohler |
| Professors: | Herbert Thielen,  Werner König |
| Semester: | Summer Semester 2023 |
| Due Date: | 30. March 2023 |

Inhalt

[Teamvorstellung: 3](#_Toc128138473)

[1. Introduction and Goals 4](#_Toc128138474)

[1.1: Requirements Overview: 4](#_Toc128138475)

[1.2: Stakeholders: 5](#_Toc128138476)

[1.3: Use Cases: 6](#_Toc128138477)

[1.4: User Stories: 6](#_Toc128138478)

[2. App Planning 7](#_Toc128138479)

[3. App Development 7](#_Toc128138480)

[Glossary: 8](#_Toc128138481)

# Team Introduction:

|  |  |  |  |
| --- | --- | --- | --- |
| Name | Rolle | Email | Matrikelnummer |
| Saadet Ibrahimova | Product Owner | inf3530@hs-worms.de |  |
| Ayoub El-Mrabet | Scrum Master | inf3822@hs-worms.de | 677623 |
| Philipp Muders | Developer | [inf3455@hs-worms.de](mailto:inf3455@hs-worms.de) | 675908 |
| Kristiyan Ivanov | Developer | inf3563@hs-worms.de | 676281 |
| Omer Guimdo-Achoungo | Developer | [inf3249@hs-worms.de](mailto:inf3249@hs-worms.de) | 674699 |
| Marc Wieland | Developer | [inf3457@hs-worms.de](mailto:inf3457@hs-worms.de) | 675904 |
| Mikulas Willaschek | Developer | [inf3825@hs-worms.de](mailto:inf3825@hs-worms.de) | 677636 |
| Nico Hofmann | Developer | inf3207@hs-worms.de | 674656 |
| Tom Brauns | Developer | [inf3621@hs-worms.de](mailto:inf3621@hs-worms.de) | 676672 |

# 1. Introduction and Goals:

The Focus of this application is to develop an application working as both an Application aswell as an Web-App. It’s supposed to enable a simplistic donation process for Guests and should have the option for Users to sign up and therefore have access to more functions and benefits on the application. The App in itself includes an calendar, a project-catalogue, aswell as Events, an donation screen and a chat.

## 1.1: Requirements Overview:

#### Functions include but are not limited to:

* Scanning a QR-Code
* The Login-Function
* The register-function
* The Change-User-Data-function
* Donate
* Continue as Guest
* Select Paymethode
* Share Donation
* Get a Donation-Receipt
* Display Events
* Display specific Event
* Create Event
* Edit Event
* Delete Event
* Display Catalogue
* Display specific project
* Create project
* Edit project
* Delete project

#### A summarization of the functional requirements would look as follows:

|  |  |  |  |
| --- | --- | --- | --- |
| ID | Requirement | Description | Reasoning |
| F-1 | Scanning a QR-Code | A User at an Event can scan a QR-Code portrayed on a Flyer | The QR-Code handles the Navigation to a specific Screen and therefore is essential |
| F-2 | A User can register | Users can register on the Application to save their data and have more functions on the Application | In order to avoid having to enter required data repeatedly, it is necessary that the User can register. |
| F-3 | A User can login | Users can choose to log in in the Application if their Account is already existing. | As an Account was already created, logging into that account saves the effort to have to enter important data repeatedly. |
| F-4 | Logged in Users can change their Data | A Logged in User can change the given Data. | Contact info might change considering the address or the Email, or the user wants to change the password. All these functions need to be provided. |
| F-5 | Donate | Users can donate |  |
| F-6 | Continue as Guest | A User can donate without being required to sign up. | Just because a User doesn’t want to sign up doesn’t mean he should be hindered from donating to a good cause. |
| F-7 |  |  |  |
| F-8 |  |  |  |
| F-9 |  |  |  |
| F-10 |  |  |  |

#### This graphic depicts the functions in accordance with the User-types. The further you move to the right, the more functions that User has access to.

## 1.2: Stakeholders:

#### As Stakeholders we consider all the people benefitting from the direct implementation and release of the application

|  |  |  |
| --- | --- | --- |
| Stakeholders | Type of Stakeholder | Reasoning |
| Donators | Primary | As we aim to make the donation process as easy as possible, we always have to think of the donators. |
| Sponsors | Primary | As with all projects, keeping an application running as well as running actual events that we use the application for requires funding. Those can be companies supporting the events, as well as organizations using it. |
| Lions | Primary | While the App isn’t directly in development for the Lions-Club, we still have to consider them or main audience group. |
| Lions Members | Secondary | As |
| Organizations | Secondary | If available, those are the organizations that handle contributing the funds to the actual problem. |
| State | Tertiary | Considering that donating involves money transactions, the state has to at least be considered for legal reasoning |
| Donation Receivers | Secondary | As they are the ones who actually benefit from the gathering of money, we should keep them in mind. |

## 1.3: Use Cases:

#### Considering the in [1.1 defined functional requirements](#_A_summarization_of) the user is left with (but not exclusively) the following use cases:

|  |  |  |
| --- | --- | --- |
| ID | Use Case | Description |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

## 1.4: User Stories:

#### The core of each app is to focus on the main functionality at first, so that an application, having the core features implemented, can be used already. To best fulfill this approach, we’ll focus on the most important Use-Cases and User-Stories.

|  |  |  |  |
| --- | --- | --- | --- |
| ID | User-Story | Function | Reference |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

# 2. App Planning

# 3. App Development

# Glossary:

User: Every type of person having access to the Application and making use of the functions.

Guest: Someone who uses the Application but does not log in / register to the page

Registered User: A registered User is a User that registered on the Application / logged into the Application. This refers to Friends, Member or Admins.

Logged In User: A logged in User is a User that is currently signed into the Application with an Email aswell as an password.

Friend: A Friend is a person that registered on the Application

Member: A Member is someone who has more rights than a Friend. They have been given the role Member by an Admin and are the organizational Users that can create Events and access the Calendar.

Admin: An Admin is a Member in its core functionality, but has further rights, such as creating new Catalogue Entries (Considered as Projects in our Application ) as well as Manage the rights of Users.

Catalogue: The Catalogue has all of the projects available portrayed in it in a List.

Projects: Projects are Categories that summarize an Catastrophy that occurred around the world. In the project, you can read up on the catastrophy, as well as see the measurements that are being taken to help. They are listed in the Catalogue

Events: Events are datebound Activities, that have the purpose of collecting money to donate for one of the Projects. They are being displayed in a List.

Donation Receivers: As Donations are gathered, the Receivers refers to the Organization or the people that benefit from that money.