

# ALFRED

Personal Interactive Assistant for Independent Living and Active Ageing



## WP3 – ALFRED Core

### D3.5.1– App Development Reference and Marketplace Integration

Deliverable Lead: ASC

Delivery Date: 09/2015

Dissemination Level: Public

Version 1.0

This deliverable provides a documentation of the first efforts being put into a well-defined app development reference for third party developers. It specifies the scope of this version and the degree of fulfilment of the requirements to be covered by the documentation. Moreover, it specifies how developers will be able to develop for the Personal Assistant Service and how to consume the shared ALFRED APIs.



Document Status	
Deliverable Lead	ASC
Internal Reviewer 1	TALK
Internal Reviewer 2	TUDA
Type	Deliverable
Work Package	WP3: ALFRED Core
ID	D3.5.1: App Development Reference and Marketplace Integration
Due Date	30.09.2015
Delivery Date	30.09.2015
Status	For Approval

## Note

*This deliverable is subject to final acceptance by the European Commission.*

## Disclaimer

*The views represented in this document only reflect the views of the authors and not the views of the European Union. The European Union is not liable for any use that may be made of the information contained in this document.*

*Furthermore, the information is provided “as is” and no guarantee or warranty is given that the information is fit for any particular purpose. The user of the information uses it at its sole risk and liability.*

## Project Partners

 <p>Ascora GmbH, Germany</p>	 <p>Atos Spain sau, Spain</p>
 <p>Worldline, Spain</p>	 <p>Charité - Universitätsmedizin Berlin - Department of Geriatrics, Germany</p>
 <p>Asociacion de Investigacion de la Industria Textil, Spain</p>	 <p>Technische Universität Darmstadt, Germany</p>
 <p>National Foundation for the Elderly, The Netherlands</p>	 <p>Talkamatic AB, Sweden</p>
 <p>E-Seniors, France</p>	 <p>TIE Nederland N.V., The Netherlands</p>
 <p>IESE Business School, Spain</p>	

## Executive Summary

ALFRED – Personal Interactive Assistant for Independent Living and Active Ageing – is a project funded by the Seventh Framework Programme of the European Commission under Grant Agreement No. 611218. It will allow older people to live longer at their own homes with the possibility to act independently and to actively participate in society.

This is achieved by providing the technological foundation for an ecosystem, which consist of an User-Driven Interaction Assistant, which is basically a Smartphone App, allowing older people to talk to ALFRED and to ask questions or define commands in order to solve day-to-day problems. It also aims at improving the social inclusion by suggesting personalized social events, taking into account their interests and their social environment. Medical staff and caretakers can access the vital signs of older people monitoring by (wearable) sensors. Finally, by using serious games, physical and cognitive Impairments will be prolonged.

In this context, third party developers are welcome to contribute to the overall ALFRED eco system.

This can be achieved by joining ALFRED on GitHub (<https://github.com/ALFREDProject>). Here, every developer will find samples latest by the end of M30. It is also planned to add a “blueprint” style repository, which can be considered a template for the ALFRED development.

Furthermore, every developer has the possibility to get access to the ALFRED Personal Assistant and the Personal Assistant Commons, a library of components needed for the development of third party applications. Both parts are necessary for the development of ALFRED applications.

Thanks to a sophisticated and unified approach, third party apps connect easily to the Personal Assistant Service, and can also use the available APIs in the same way.

Special preparations are needed for CADE, the Context-Aware Dialoge Engine, which allows the user to communicate with the ALFRED system via voice interaction. Apps can offer this service to the end user, but special preparations have to be performed to use this functionality in the app. It is necessary to provide a Dialogue-Domain Description (DDD), which will then be installed on the CADE backend server, and will allow the engine to make decisions and offer help on the context given.

Finally, the marketplace integration will allow the developer to distribute his or her app. As the marketplace requires a more mature status of the overall system to be fully integrated, it will be described in detail in the upcoming deliverable D3.5.1.

D3.5.1 - App Development Reference and Marketplace Integration	Document Version: 1.0	Date: 2015-09-30	Status: For Approval	Page: 4 / 15
<a href="http://www.alfred.eu/">http://www.alfred.eu/</a>	Copyright © ALFRED Project Consortium. All Rights Reserved. Grant Agreement No.: 611218			

## Table of Contents

1	Introduction .....	7
1.1	ALFRED Project Overview .....	7
1.2	Deliverable Purpose, Scope and Context.....	8
1.3	Document Status and Target Audience .....	8
1.4	Abbreviations and Glossary.....	8
1.5	Document Structure.....	8
2	Context and Scope .....	9
3	App Development Reference.....	10
3.1	The Github-Repositories.....	10
3.2	Getting the Personal Assistant .....	10
3.3	The Personal Assistant Service.....	10
3.4	Use the implementations .....	12
3.5	Integrating CADE in a third party application.....	12
4	Marketplace Integration .....	14
5	Summary .....	15

## List of Figures and Listings

### List of Figures

Figure 1: Overview of the Personal Assistant Architecture .....	11
Figure 2: Sequence diagram of the CADE interaction for third party apps. ....	13

### Listings

Listing 1: Connecting to the Personal Assistant Service .....	11
Listing 2: Calling a wrapper library in your project .....	12

# 1 Introduction

ALFRED – Personal Interactive Assistant for Independent Living and Active Ageing – is a project funded by the Seventh Framework Programme of the European Commission under Grant Agreement No. 611218. It will allow older people to live longer at their own homes with the possibility to act independently and to actively participate in society by providing the technological foundation for an ecosystem consisting of four pillars:

- **User-Driven Interaction Assistant** to allow older people to talk to ALFRED and to ask questions or define commands in order to solve day-to-day problems.
- **Personalized Social Inclusion** by suggesting social events to older people, taking into account their interests and their social environment.
- A more **Effective & Personalized Care** by allowing medical staff and caretakers to access the vital signs of older people monitored by (wearable) sensors.
- **Physical & Cognitive Impairments Prevention** by way of serious games that help the users to maintain and possibly even improve their physical and cognitive capabilities.

This deliverable provides a documentation of the first efforts being put into a well-defined app development reference for third party developers. It specifies the scope of this version and the degree of fulfilment of the requirements to be covered by the documentation. Moreover, it specifies how developers will be able to develop for the Personal Assistant Service and how to consume the shared ALFRED APIs.

## 1.1 ALFRED Project Overview

One of the main problems of western societies is the increasing isolation of older people, who do not actively participate in society either because of missing social interactions or because of age-related impairments (physical or cognitive). The outcomes of the ALFRED project will help to overcome this problem with an interactive virtual butler (a smartphone application also called ALFRED) for older people, which is fully voice controlled.

The ALFRED project is wrapped around the following main objectives:

- To empower older people to live independently for longer by delivering a virtual butler with seamless support for tasks in and outside the home. This virtual butler (the ALFRED app) aims for a very high end-user acceptance by using a fully voice controlled and non-technical user interface.
- To prevent age-related physical and cognitive impairments with the help of personalized serious games.
- To foster active participation in society for the ageing population by suggesting and managing events and social contacts.
- And finally, to improve caring by offering direct access to vital signs for carers and other medical staff as well as alerting in case of emergencies. The data is collected by unobtrusive wearable sensors monitoring the vital signs of ALFRED's users.

To achieve its goals, the project ALFRED conducts original research from a user centred perspective and applies technologies from the fields of Ubiquitous Computing, Big Data, Serious Gaming, the Semantic Web, Cyber Physical Systems, the Internet of Things, the

D3.5.1 - App Development Reference and Marketplace Integration	Document Version: 1.0	Date: 2015-09-30	Status: For Approval	Page: 7 / 15
<a href="http://www.alfred.eu/">http://www.alfred.eu/</a>		Copyright © ALFRED Project Consortium. All Rights Reserved. Grant Agreement No.: 611218		

Internet of Services, and Human-Computer Interaction. For more information, please refer to the project website at <http://www.alfred.eu>.

## 1.2 Deliverable Purpose, Scope and Context

The purpose of this deliverable is to describe the current grade of advance in the development in the guidelines and development references to allow third party developer to create ALFRED apps.

## 1.3 Document Status and Target Audience

This deliverable is qualified as public in the Description of Work (DoW). It should a helpful insight to understand the ALFRED System..

## 1.4 Abbreviations and Glossary

A definition of common terms and roles related to the realization of the ALFRED project as well as a list of abbreviations is available in the supplementary document “Supplement: Abbreviations and Glossary”, which is provided in addition to this deliverable. Further information can be found at <http://www.alfred.eu>.

## 1.5 Document Structure

This deliverable is broken down into the following sections:

- **Chapter 1** provides an introduction for this deliverable including a general overview of the project, and outlines the purpose, scope, context, status, and target audience of this deliverable.
- **Chapter 2** describes the main scope of the task by providing general information and a scope of the prototype itself.
- **Chapter 3** outlines the steps needed to work with the Personal Assistant Service.
- **Chapter 4** describes the steps needed to integrate the app into the ALFREDO marketplace.
- Finally, **Chapter 5** summarizes the aforementioned content.

D3.5.1 - App Development Reference and Marketplace Integration	Document Version: 1.0	Date: 2015-09-30	Status: For Approval	Page: 8 / 15
<a href="http://www.alfred.eu/">http://www.alfred.eu/</a>	Copyright © ALFRED Project Consortium. All Rights Reserved. Grant Agreement No.: 611218			



## 2 Context and Scope

The App Development Reference is targeted at developers for the ALFRED eco system. It's main focus is build around the need for better documentation and attracting new developers to the overall system.

It will help the developer to benefit from the already pre-defined APIs of the ALFRED eco system, and allows for better coding, thus raising the overall application quality in the ALFREDO Marketplace.

D3.5.1 - App Development Reference and Marketplace Integration	Document Version: 1.0	Date: 2015-09-30	Status: For Approval	Page: 9 / 15
<a href="http://www.alfred.eu/">http://www.alfred.eu/</a>	Copyright © ALFRED Project Consortium. All Rights Reserved. Grant Agreement No.: 611218			

### 3 App Development Reference

Developers should consider getting their hands on the sample application, as stated in D3.2.2, to get the best in depth information on how to handle the communication with the first prototype.

However, the following sections will give a sophisticated overview on the overall API.

#### 3.1 The Github-Repositories

For a better experience for third-party-developers, an organization has been created on github.com, a web-based Git repository hosting service, which should host all open, ALFRED-associated projects. These projects are supposed to offer a reference for developer. During the project, more and more projects should be added to the organization.

The ALFRED organization can be found at <https://github.com/ALFREDProject>.

For D3.5.2, at least two apps should be made available on Github, showing different ways of interacting with different parts of the ALFRED system. Third party developer will then be able to use the apps as samples for there own projects.

It is also planned to add a “blueprint” repository to the ALFRED project on github. Developer will then be able to use this template as a base for their own projects. It will have different parts of the ALFRED system already integrated, like the call of the service, and the initialization of CADE. These are two key concepts, which can then be easily applied on the other APIs available.

#### 3.2 Getting the Personal Assistant

To work with the Personal Assistant, it is mandatory to get the Personal Assistant App on the development device, because all services that the ALFRED ecosystem offers are made available through the Personal Assistant Service.

The Personal Assistant App can be obtained through the ALFRED CI server (the latest version), or is attached to this document. It will run on Android Phones starting with Android targeting SDK version 16 (that is, Android 4.1). The reference device is a Motorola Nexus 5.

Sideloaded apps need to be allowed for the Personal Assistant to work. Besides of that, a working WiFi or mobile plan is required for the speech interaction to work.

#### 3.3 The Personal Assistant Service

Figure 1: Overview of the Personal Assistant Architecture shows the detailed view for the architecture of the Personal Assistant.

D3.5.1 - App Development Reference and Marketplace Integration	Document Version: 1.0	Date: 2015-09-30	Status: For Approval	Page: 10 / 15
<a href="http://www.alfred.eu/">http://www.alfred.eu/</a>	Copyright © ALFRED Project Consortium. All Rights Reserved. Grant Agreement No.: 611218			

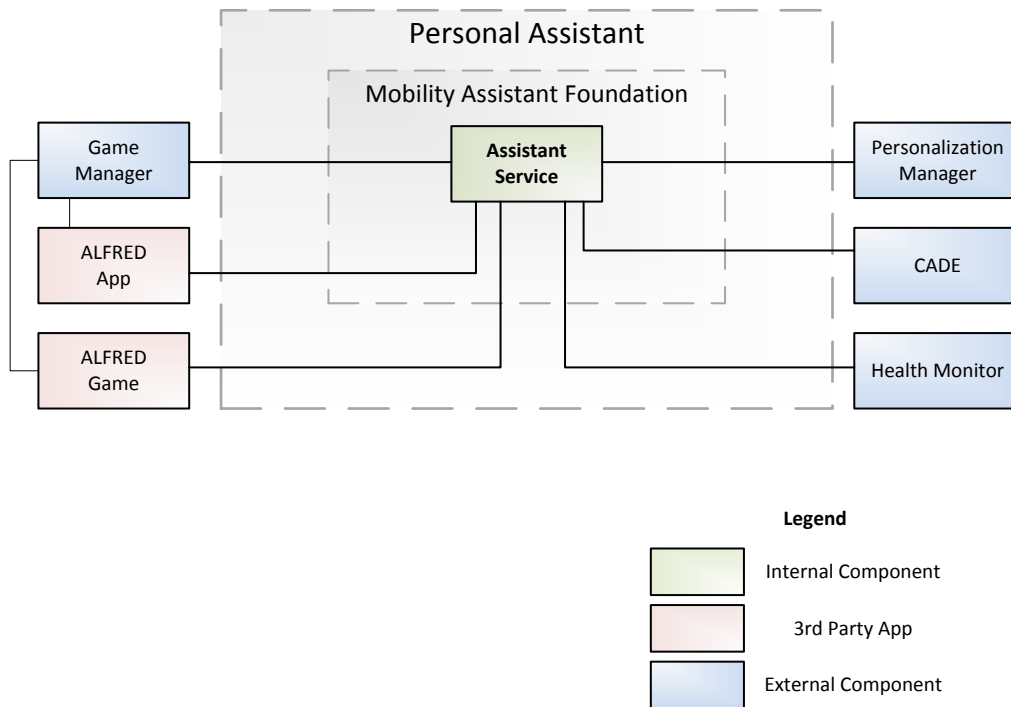


Figure 1: Overview of the Personal Assistant Architecture

An app that consumes the services of the ALFRED ecosystem, needs to include the `personalassistantshared-debug.aar` which is part of the binary package you get with this deliverable, or can be found in the Jenkins CI in the job “PersonalAssistantCommons”.

These Commons-library might as well be made available via `github.com`, which will be discussed in the upcoming deliverable D3.5.2.

To actually use and connect to the Personal Assistant Service, very little code is required, as described in Listing 1: Connecting to the Personal Assistant Service

Listing 1: Connecting to the Personal Assistant Service

```

personalAssistant = new PersonalAssistant(this);

personalAssistant.setOnPersonalAssistantConnectionListener(new
PersonalAssistantConnection() {
    @Override
    public void OnConnected() {
        // Do some stuff
    }

    @Override
    public void OnDisconnected() {
        // Do some cleanup stuff
    }
});

personalAssistant.Init();
  
```

The `PersonalAssistant` class needs a context provided in the constructor. The only event available is the `setOnPersonalAssistantConnectionListener`, which offers the two methods `OnConnected` and `OnDisconnected`.

`OnConnected` is called after the library has established the connection to the Personal Assistant Service. In the `OnConnected` method, it is possible to initialize several APIs like the `GameManager` or `CADE`. The `OnDisconnected` method is used likewise for cleaning up the used instances.

This is the overall initialization workflow for the Personal Assistant Service.

Finally, the `Init` method has to be called. It will do the binding to the service itself.

### 3.4 Use the implementations

Interacting with the available ALFRED APIs is straightforward, as they are very convenient to use.

Listing 2: Calling a n ALFRED Service

```
personalAssistant = new PersonalAssistant(this);

personalAssistant.setOnPersonalAssistantConnectionListener(new
PersonalAssistantConnection() {
    @Override
    public void OnConnected() {
        // Do some stuff
        cade = new Cade(personalAssistant.getMessenger());
    }

    @Override
    public void OnDisconnected() {
        // Do some cleanup stuff
    }
});

personalAssistant.Init();
```

Listing 2: Calling a n ALFRED Service shows how to integrate a wrapper library in a third party application. In this case, we will react on the `OnConnected` event, and will create a new instance of the Context-Aware Dialogue Engine (CADE). This object can then be used through-out the application lifetime.

In general, every wrapper library will need an injected dependency, an instance of the `Messenger` class, which gets created by the `PersonalAssistant` object.

### 3.5 Integrating CADE in a Third Party Application

Every API is called in a similar way. However, if the integration of CADE has been considered for an app, some additional tasks for the integration of the overall voice command infrastructure have to be performed.

First, every developer needs to provide a Dialogue Domain Description (DDD), as described in D4.1.2. This description will then be installed into the CADE Session Manager and CADE Backend, which is running on an ALFRED server instance.

Additionally, the developer will need to integrate an additional IPC channel, which is based on Intents. The Personal Assistant Service will send a special Intent to the calling application. This intent has the plaintext command as payload, and should be handled by the app accordingly. Figure 2: Sequence Diagram of the CADE Interaction for Third Party Apps shows a complete run of the speech interaction.

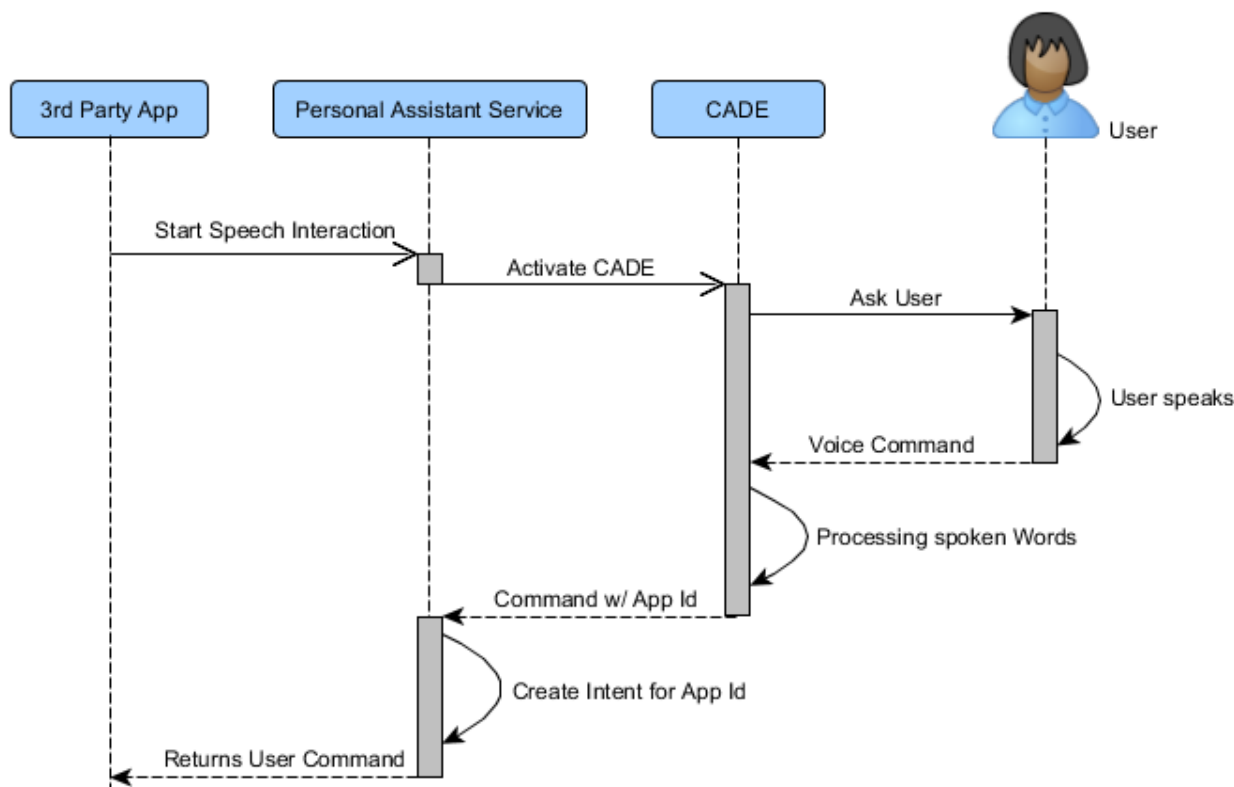


Figure 2: Sequence Diagram of the CADE Interaction for Third Party Apps

The developer needs to send a command back to the Personal Assistant, that he has taken care of the response from the Personal Assistant Service.

The first draft names this method `CommandConfirmed`, although the naming might be subject to change before D3.5.2.

## 4 Marketplace Integration

Because the integration of the ALFREDO marketplace is directly bound to the overall progress of the Mobile Assistant Foundation / Personal Assistant Service, its usage and the necessary prerequisites will be outlined in detail in the deliverable D3.5.2.

D3.5.1 - App Development Reference and Marketplace Integration	Document Version: 1.0	Date: 2015-09-30	Status: For Approval	Page: 14 / 15
<a href="http://www.alfred.eu/">http://www.alfred.eu/</a>	Copyright © ALFRED Project Consortium. All Rights Reserved. Grant Agreement No.: 611218			

## 5 Summary

This deliverable has presented and described the overall approach established so far to get developers to work with the ALFRED ecosystem. It is mandatory for developers to understand the need for a unified approach, as the ALFRED system presents it.

Furthermore, the deliverable showed that effort has been put into kickstarting the public visibility from a developer perspective by choosing Github as a main platform for public available projects.

Finally, it showed how to integrate the Personal Assistant Service itself into the app, and how to integrate the ALFRED APIs.

The upcoming deliverable D3.5.2 will then mainly focus on the overall integration of ALFRED apps into the ALFREDO Marketplace.

D3.5.1 - App Development Reference and Marketplace Integration	Document Version: 1.0	Date: 2015-09-30	Status: For Approval	Page: 15 / 15
<a href="http://www.alfred.eu/">http://www.alfred.eu/</a>	Copyright © ALFRED Project Consortium. All Rights Reserved. Grant Agreement No.: 611218			