AAL Programme

SUCCESS - **SU**ccessful **C**aregiver **C**ommunication
and **E**veryday **S**ituation **S**upport in dementia care

PROJECT IDENTIFICATION	
PROJECT NUMBER	AAL-2016-089
DURATION	1st March 2017 – 29th February 2020
COORDINATOR	Markus Garschall
COORDINATOR ORGANIZATION	AIT Austrian Institute of Technology GmbH
WEBSITE	www.success-aal.eu

DOCUMENT IDENTIFICATION	
DELIVERABLE ID	D4.2 Security and privacy infrastructure specification
RELEASE NUMBER / DATE	v1.0 / 09.08.2017
CHECKED AND RELEASED BY	Markus Garschall (AIT)

KEY INFORMATION FROM ‘DESCRIPTION OF WORK’	
DELIVERABLE DESCRIPTION	This report contains a security evaluation of the SUCCESS system architecture and the resulting measures and specifications to be taken in the architecture to mitigate security risks.
DISSEMINATION LEVEL	Public
DELIVERABLE TYPE	Report
ORIGINAL DUE DATE	Project month 5

AUTHORSHIP & REVIEWER INFORMATION	
EDITOR	Hannes Zach (EXT), Jakob Hatzl (EXT)
PARTNERS CONTRIBUTING	EXT, SIL, UCY
REVIEWED BY	Dimitrios Ntalaperas (SIL)

ABBREVIATIONS

ABBREVIATIONS	DESCRIPTION
REST	Representational State Transfer
PwD	Person with Dementia
YAML	YAML Ain't Markup Language
JSON	JavaScript Object Notation

TABLE OF CONTENTS

EXECUTIVE SUMMARY

The present report documents the security and privacy infrastructure for the overall SUCCESS system to ensure data security and data integrity where sensitive data is processed.

Based on the system architecture, described in *D4.1: Functional specification and integrated architecture report*, security risks for each of the main components and their interfaces were identified. Each risk was quantified by impact and probability to get a prioritization for security requirements. Based on the risk quantification, mitigation strategies for each risk were identified and documented. Those mitigation strategies will be incorporated in the agile development of the system architecture.

As the Use-Cases and system architecture are still evolving, this report will be updated accordingly when new releases of the relevant deliverables (mainly *D4.1: Functional specification and integrated architecture report*) are available.

1 ABOUT THIS DOCUMENT

1.1 ROLE OF THE DELIVERABLE

This deliverable describes the security and privacy infrastructure for the overall SUCCESS system to ensure data security and data integrity where sensitive data is processed. The content of this deliverable is based on a three step process to identify, quantify and mitigate security risks for the SUCCESS software. The three steps of the process are:

- an analysis of the system architecture with security requirements in mind to identify relevant security risks. This analysis is mainly based on the system architecture described in *D4.1: Functional specification and integrated architecture report*.
- the quantification of identified security risks following the schematic DREAD approach.
- the identification and documentation of strategies to mitigate the identified and prioritized risks.

Concerns regarding the security and privacy infrastructure in connection with the physical deployment of the SUCCESS application (hosting, server housing, firewalls, routers, …) are not covered in this release of the deliverable. A further release of this deliverable will cover this topic additionally after the deployment view of the architecture is clarified in the next release of *D4.1: Functional specification and integrated architecture report*.

1.2 RELATIONSHIP TO OTHER SUCCESS DELIVERABLES

The deliverable is related to the following SUCCESS deliverables:

DELIVERABLE	RELATION
D4.1	Functional specification and integrated architecture report: this deliverable describes the overall SUCCESS system architecture and functions and is the basis for the security related considerations outlined in this deliverable 4.2

1.3 STRUCTURE OF THIS DOCUMENT

This deliverable is structured in three main chapters, covering the three step process described in section 1.1.

Chapter 2 covers the system analysis and recapitulates the architecture described in *D4.1: Functional specification and integrated architecture report* from the viewpoint of security and privacy. The security relevant parts of each main component are outlined and described in detail.

Chapter 3 lists the risks identified through the analysis in chapter 2. Each risk is quantified following the DREAD principle that measures risks by impact and probability.

Chapter 4 lists mitigation strategies for the risks identified and quantified in chapter 3 and acts as a reference point in the further implementation of the system.

2 SYSTEM ANALYSIS

The system architecture of SUCCESS is designed in a modular way, where components within the system are loosely coupled to reduce dependencies between all components (see Figure 1). In addition, each component will also consist of a standalone module, and will thus be able to be deployed as either an application or a micro service.

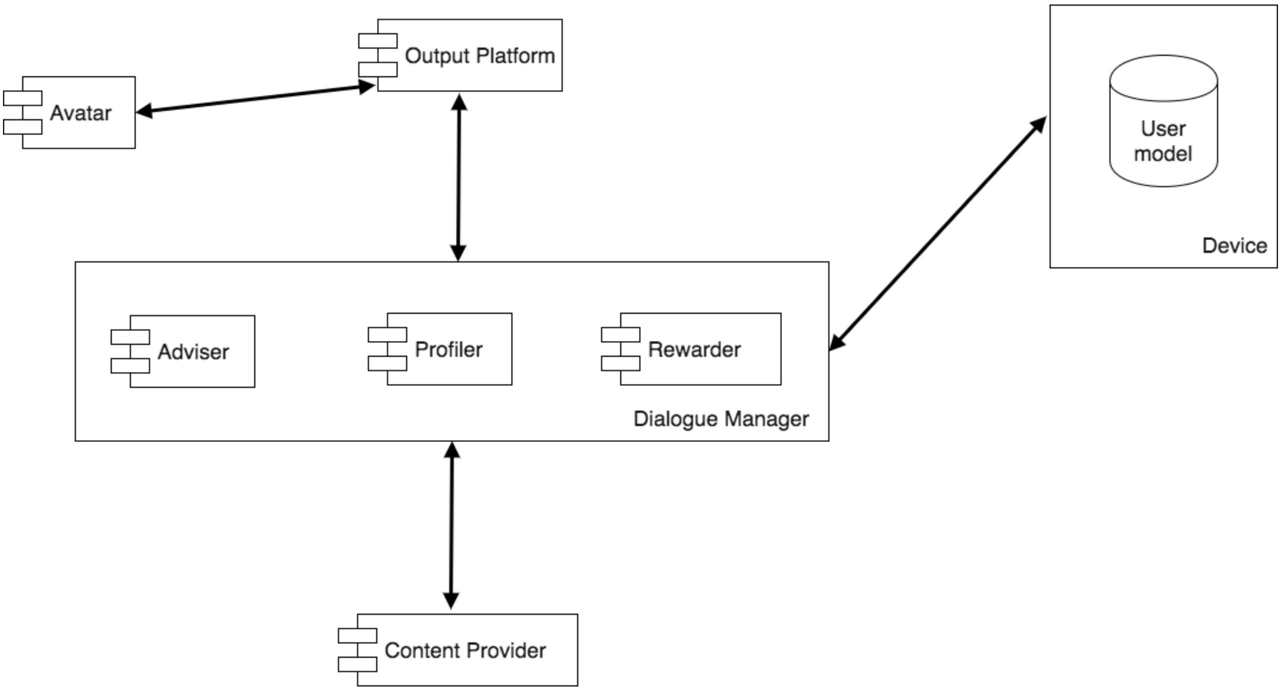

Figure 1 Architectural Design of the SUCCESS system.

Figure 2 provides a depiction of the level 1 Component Diagram, where a more detailed interconnection of the application components is presented. Each represented component is a modular part of the application, whose behavior is defined by its provided and required interfaces. An Interface is a specification of behavior that implementer components agree to meet. An assembly connector is used in order to bridge a component's required interface with the provided interface of another component.

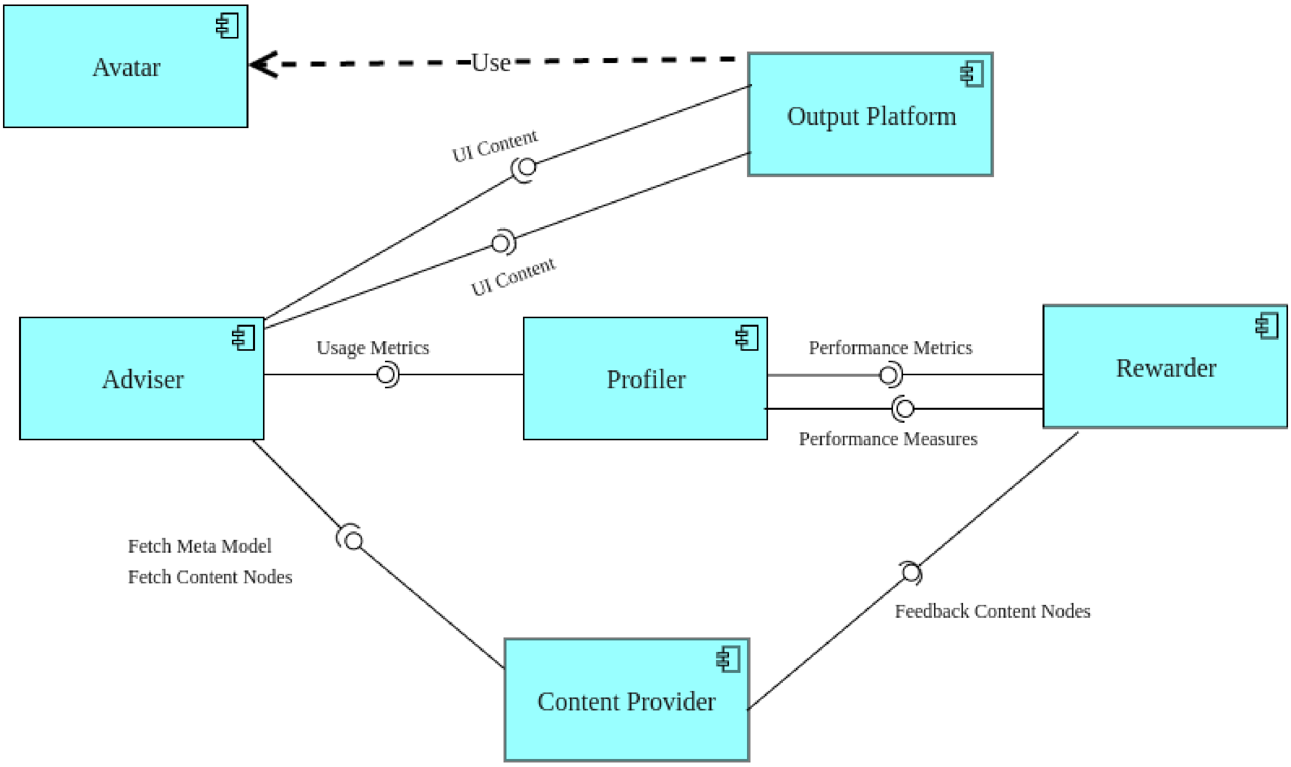

Figure 2 Development View of the Components

2.1 OUTPUT PLATFORM

The Output Platform represents the main user frontend with which the users interact directly with the system. The main purpose of the Output Platform is the rendering of the user interface that will allow the user to interact with the system and use the services provided by SUCCESS

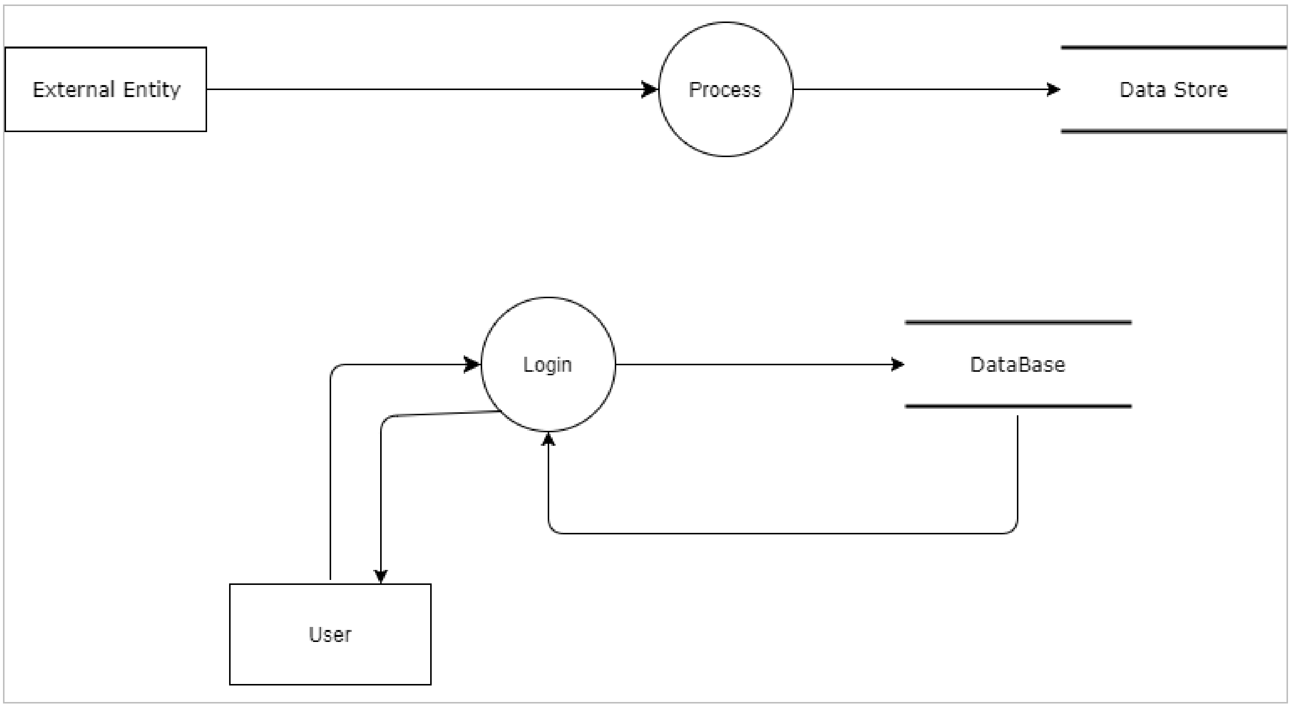

Figure 3: Level 0 Data Flow Diagram for the Output Platform

From the DFD in Figure 3, we can identify that the **login process** is one of the **main processes** within the output platform. The detailed view of the output platform is visualized in Figure 4:

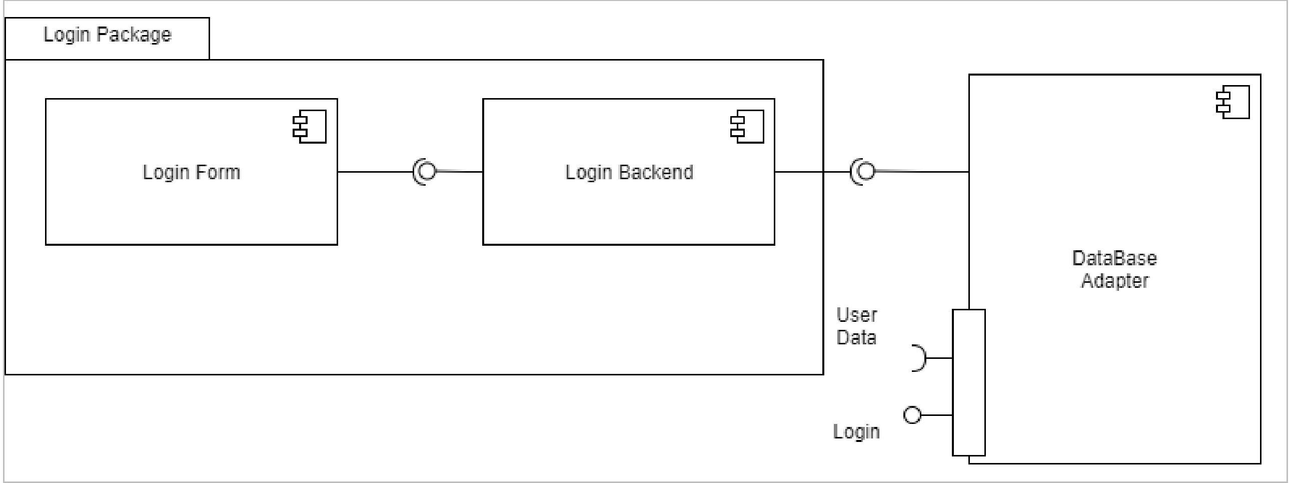

Figure 4: Level 0 Component Diagram for the Output Platform

The login form is the interface to the user and receives input, which has to be **classified as untrusted**. Through the login backend, this (untrusted) data is used to query the database.

The Interfaces of the Output Platform with other modules are depicted in Table 1. The Profiler needs to communicate the output of the questionnaires to the Output Platform and provides the usage metrics needed by the Adviser. In the data level, the Profiler needs to interface with the User Model in order to be able to retrieve information about the user.

Interface	Description	Interfacing Component	Type	Format
Login	User can login to the application	Google Sign-In	Google API	HTTPS
Sync with cloud	User preferences can be stored and synchronized throughout different devices.	Google Sign-In	Google Drive REST API	HTTPS
User Actions	User can navigate through the application to retrieve information.	Adviser	Java internal	Java Objects/binary

Table 1 Communication Interface of the Output Platform

2.2 PROFILER

The Profiler component is one of the three components realising the dialogue management in the SUCCESS solution. The main purpose of the Profiler is to administer data about the user in a user model. The Profiler does this through continuously recording of data during system usage, as well as through initiating active inquiry. The only input to the system is direct user input.

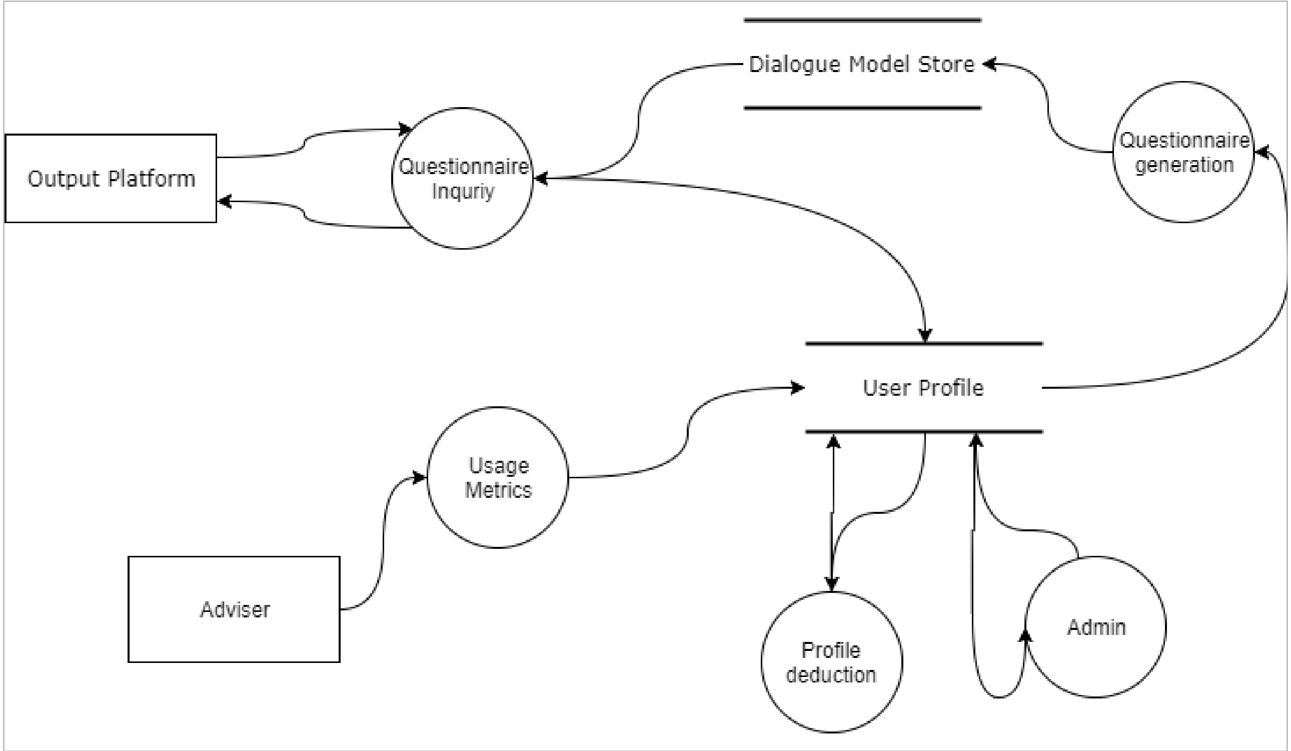

Figure 5: Level 0 Data Flow Diagram for the Profiler

Based on Figure 5, the following is considered as relevant for the security model:

The profiler component

- is in constant contact with the advisor component and receives data from the adviser, which is then used to update the user model,
- accesses and updates the user profile / user model,
- accesses and updates the dialogue model store and
- is responsible for continuously recording of data during system usage.

The Interfaces of the Profiler with other modules are depicted in Table 2. The Profiler provides interfaces for general application activity as well as interfaces for both adviser and dialogue specific activities. In the data level, the Profiler needs to interface with the Profiler Database in order to be able to retrieve information about the user.

Interface	Description	Interfacing Component	Type	Format
IProfiler	Profiler interface for general application activity	Output Platform	Java internal	Java Objects
IAdvisorProfiler	Profiler interface for adviser specific activities	Adviser Output Platform	Java internal	void
IDialogueProfiler	Profiler interface for dialogue specific activities	Adviser/Dialogue	Java internal	Java Objects

ProfilerDatabase	Database interface to read and write user profile data	ProfilerDatabase	Java internal	Java Objects

Table 2 Communication Interface of the Profiler

2.3 REWARDER

Together with the adviser and the profiler component, the rewarder component is realizing the dialogue management within SUCCESS. The main purpose of the rewarder component is to choose or generate feedback to the user. It does so by choosing appropriate feedback for performance measures from the system (i.e. derived from Profiler data). The rewarder utilises adequate and meaningful content to increase motivation and engagement with the trainings.

The rewarded has not been implemented yet and will be available after the next iteration of the development process.

Table 3 lists the interfaces the Rewarder shares with some of the components of the SUCCESS solution. Due to the fact that this component is under development the specific interfaces are not defined yet, but the description of the interfaces that are intended to be used is provided.

Interface	Description	Interfacing Component	Type	Format
To be defined	Send Output Requests for Feedback	Output Platform		
	Receive User Interaction data			
	Receive feedback content nodes	Content Provider		
	Send Performance Measures	Profiler		
	Receive Performance Metrics			

Table 3 Communication Interfaces of the Rewarder

2.4 ADVISER

The adviser is the last part of the dialogue manager and its main purpose is to produce an advising entity that generates a dialogue like interaction from content nodes that contain dialogue models and the structure of the Meta model. The dialogue like interaction is expressed textually and visually and audibly supported by the avatar component.

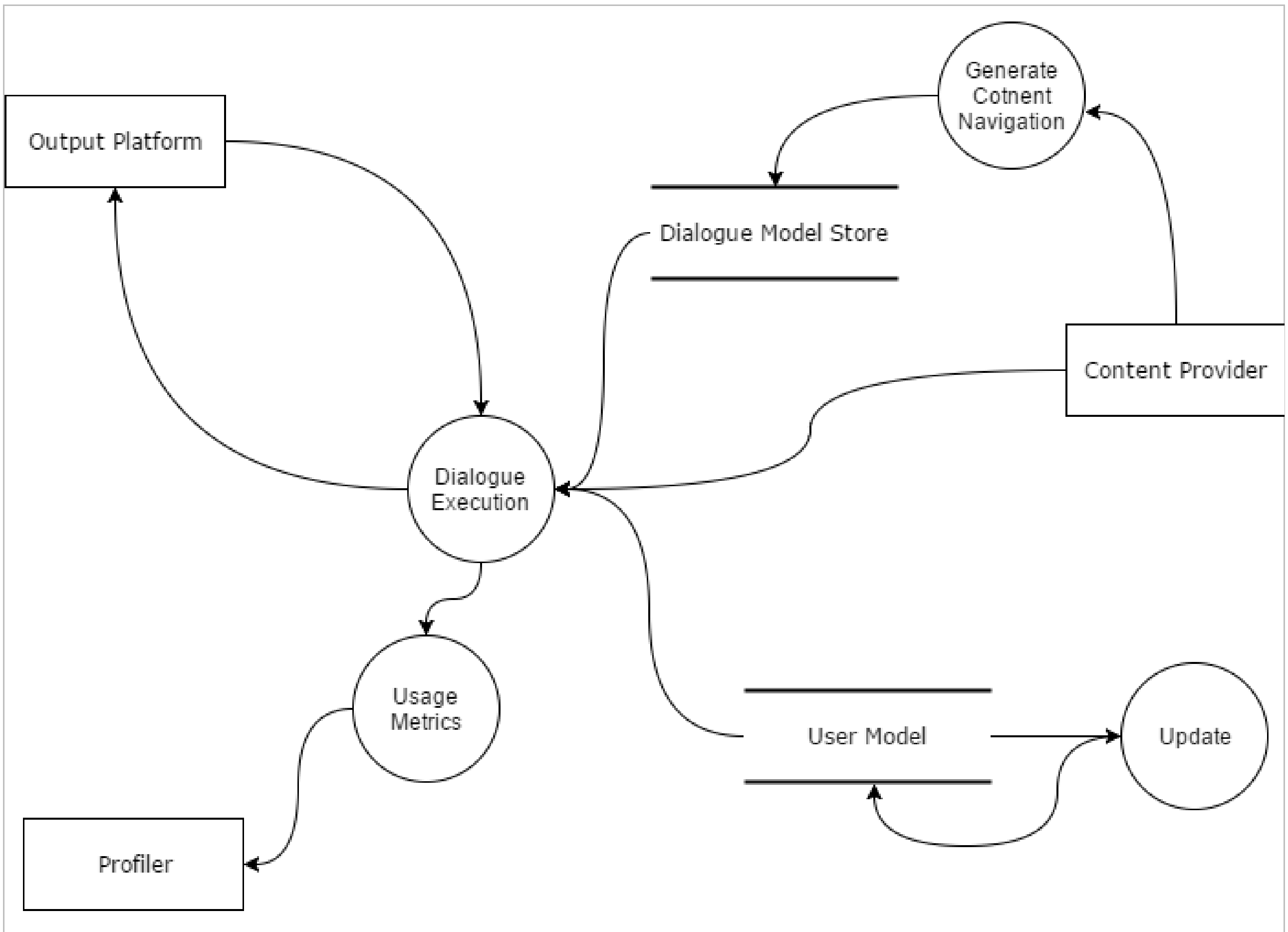

Figure 6 Level 0 Data Flow Diagram for the Adviser

Based on Figure 6, the following is considered as relevant behaviour for the security model: The adviser

- accesses the meta model (only read access),
- receives data from the output platform,
- reads and updates the user profile, and
- executes content queries for the content provider.

Relevant communication interfaces for the adviser are summarized in Table 4. According to the current state of the architecture, most interfaces are internal Java where interfaces are declared inside another interface or class and exchange Java objects. The Adviser communicates with the Output Platform by generating real time interaction instructions for the Output platform's avatar component.

Interface	Description	Interfacing Component	Type	Format
IAdvisor	Provide navigation/content tree and select navigation/content nodes	Output Platform	Java internal	Java Objects
	Query binary resources	Output Platform	Java internal	binary

IAdvisorCallback	Callback interface enabling the Adviser to push information to components using the Adviser	Output Platform	Java internal	Java Objects
IDialogueCallback	Callback interface enabling the Dialogue to push information to components using the Dialogue	Output Platform	Java internal	Java Objects

Table 4 Communication Interface of the Adviser

2.5 CONTENT PROVIDER

The content provider is the backend component that is responsible for managing and providing the contents that are presented to the Users. The CP communicates with the Content Repository (CR), which stores the content of SUCCESS in a hierarchical way. In addition, the CP communicates with the Dialogue Manager and converts and provides the required resources from the CR. The Content Provider is the only component that accesses the content repository to save, update and retrieve contents that are presented to the users.

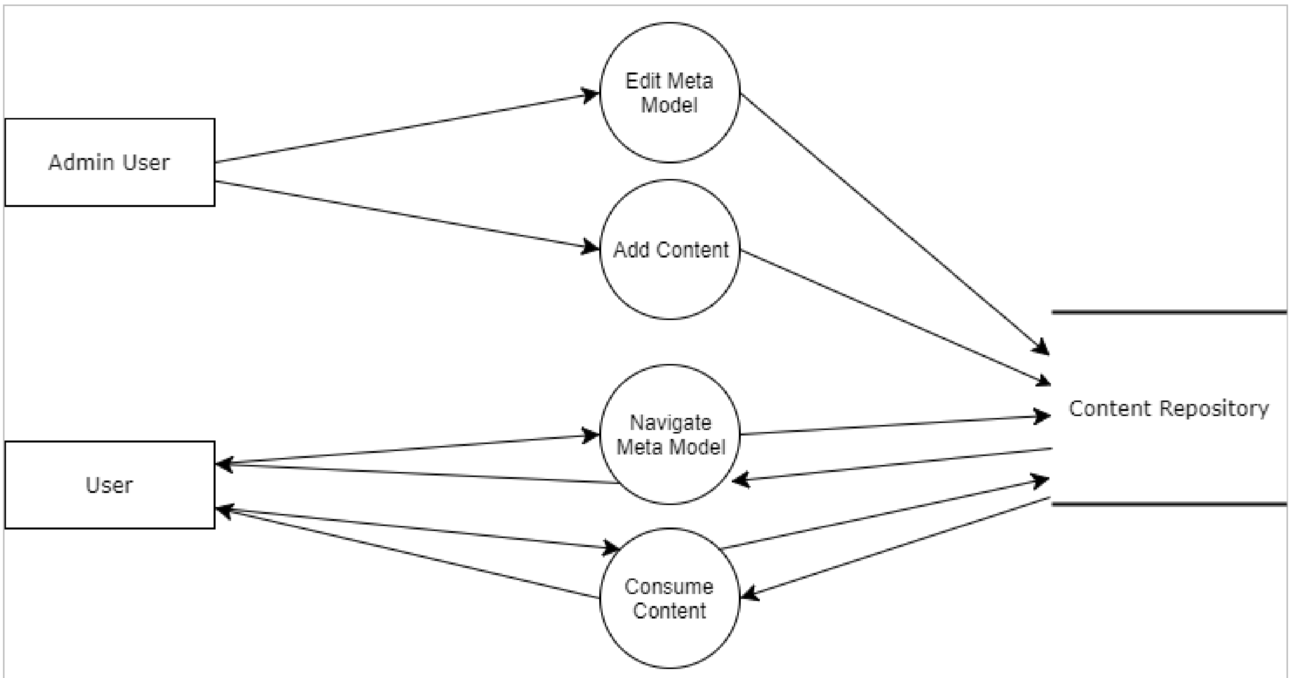

Figure 7 Level 0 Data Flow Diagram for the Content Provider

Based on Figure 7, the following information is considered as relevant for the security model: The content provider

- has full access to the repository, and
- handles the access of different users and user groups to the repository.

Relevant communication interfaces for the content provider are summarized in Table 5. The Meta Model API which is used to transverse the Content Tree that is shared with three components (Rewarder, Adviser and Output Platform) and the data will be in YAML. The actual contents of content nodes will be transmitted to the Adviser in binary format and the data from the Content Repository will be in JSON format.

Interface	Description	Interfacing Component	Type	Format
Meta Model API	This interface will allow interaction with the Content Tree (traversing the tree and node structure)	Rewarder Adviser Output Platform	Java internal	YAML
Content API	This interface will provide binary content from specific content nodes	Adviser	Java internal	binary
Repository API	This interface will provide data (content, content tree) from the Content Repository	Content Provider	REST	JSON binary

Table 5 Communication Interfaces of the Content Provider

2.6 AVATAR

The purpose of the avatar component is to provide a more natural way of interaction with the SUCCESS system. The avatar is used in lecture sessions to guide the user through dialogue interactions or acts as an interaction partner in roleplay sessions.

The avatar is implemented in Unity and wrapped in an Android fragment activity, implementing an interface that defines how other components can interact with the Avatar component.

Table 6 lists the interfaces the Avatar shares with the Output Platform. The interfaces are intended to provide the actual interaction among the Avatar and the Output Platform and based on the current state of development, most interfaces are internal Java and exchange Java objects.

Interface	Description	Interfacing Component	Type	Format
IUnityPlayerFragment	Provides the means to interact with the Avatar	Output Platform	Java internal	Java Objects
IAvatarSpeechOutput-Listener	Callback interface enabling the parent Activity to react on currently spoken text by the Avatar	Output Platform	Java internal	Java Objects

Table 6 Communication Interfaces of the Avatar

3 SECURITY MODEL

This chapter provides a risk assessment model and therefore contains a list of threats based on the analysis of each component together with an evaluation to measure the severity and impact of each single threat. The list of threats is summarized in Table 7, while the reasoning behind the threat evaluation is described subsequently. The categories to measure the threat is based on the DREAD principle [2] and include:

Damage Potential (DP), Reproducibility (R), Exploitability (E), Affected Users (A) and Discoverability (D)

The **risk value** is then calculated by **Impact * Probability** →(DP+A) * (R+E+D)

As further described in [3], for each category, a number between 0 to 5 indicates how serious the risk is. For each category, this means:

- Damage Potential: 0 = No risk, 5 = complete system or data destruction
- Reproducibility: 0 = Very hard or impossible, even for administrators of the application,
  3 = Just a web browser and the address bar is sufficient, without authentication
- Exploitability: 0 = Advanced programming and networking knowledge, with custom or advanced attack tools
  4 = Just a web browser is sufficient
- Affected Users: 0 = none, 5 = all users
- Discoverability: 0 = Very hard to impossible; requires source code or administrative access,
  3 = The information is visible in the web browser address bar or in a form.

Resulting from multiplication of the factors, the following risk categories exist:

- Low: 6-15
- Moderate: 16-40
- High > 41

ID	Threat	DP	R	E	A	DC	Risk Value
1. Output platform							
1.1	Critical user data transferred between the output platform and all connected components is not handled confidentially and can be intercepted by an attacker	5	3	3	5	3	**88**
1.2	The password of a user is directly stored as plaintext and can be compromised by an attacker	5	3	4	5	2	**88**
1.3	The hashing algorithm is outdated which allows the attacker to draw conclusions to the users password	3	2	3	5	1	**45**
1.4	An attacker is able to use brute force to gain unauthorized access to a user account	5	3	4	5	3	**96**
1.5	The password of a user is not complex enough and can easily be guessed	5	1	4	2	3	**54**
1.6	An attacker uses social engineering to convince an administrator to reset the user password	3	3	4	2	3	**54**
1.7	After an incorrect login try, the system tells too much information about the wrong input	3	3	4	2	3	**54**
1.8	Attacks on the login functionality cannot be reproduced, therefore attackers cannot be spotted	4	3	4	5	3	**84**
1.9	Session Tokens are weakly generated and can be guessed by an attacker	5	2	3	5	3	**77**
1.10	Session Tokens can be read in the HTTP request by an attacker	5	3	3	5	3	**88**
2. Profiler							
2.1	Because single resources are not protected by requiring authentication, data can leak	4	2	3	5	2	**60**
2.2	An attacker successfully conducts a privilege escalation attack based on flaws within the systems authorization management	4	2	2	5	2	**54**
2.3	An attacker finds a way to infiltrate malicious data into the system hidden as harmless user input	5	1	2	5	1	**48**
2.4	Data which seems is coming from another component will be regarded as trusted but is in fact malicious data sent by an attacker	4	2	2	5	2	**54**
3. Rewarder							
3.1	An attacker finds a way to attack the data store and alters the stored data	5	1	1	5	1	**42**
3.2	Input coming from the data store is malicious code from an attacker to manipulate data within the data store	5	1	1	5	1	**42**

3.3	Possible attacks on the user profile do not appear because no logging mechanism are in place	4	3	3	5	3	**77**
4. Adviser							
4.1	Because of an attacker or another anomaly in the system, data within the data model is lost	5	2	3	5	3	**77**
4.2	An attacker is able to alter data within the data model because there are no restrictions on possible operations on the data store	5	2	3	5	3	**77**
5. Content provider							
5.1	Data within the data repository is altered or deleted unintendedly	5	2	3	5	3	**77**
5.2	An attacker is able to alter data within the content repository like an administrator without having the necessary permissions	5	2	2	5	2	**63**
5.3	Altering of the content repository cannot be tracked successfully	4	3	3	5	2	**70**
5.4	An attacker from outside the trusted environment is able to access the admin area	5	3	3	5	3	**88**
6 Avatar							
6.1	An attacker is able to modify the output sent to the Avatar fragment.	4	2	2	1	3	**36**

Table 7 Identified Threads of SUCCESS

Reasoning behind the rating of threats

- Whenever (user) input enters the system, there is a high potential that the data does not correspond to the format or content, expected by the developer. Therefore, all interfaces to the user expose a high risk to the system.

- Because in SUCCESS different user roles are incorporated within the whole process, authentication and authorization plays a key role for the security model.

- Within SUCCESS, different components access the same repositories (User Model), which makes tractability and logging of data access important and offers several potential attack surfaces which is the reason for a high security risk in all components with access to data stores.

- Whenever data is sent between different components there is a possible chance to intercept the communication. Because of the micro service architecture of SUCCESS, communication channels between components are considered as serious risk to the system.

4 THREAT MITIGATION AND VALIDATION

This chapter expands on the threats identified in chapter 3 and provides mitigation strategies which should be realized within the further development process and should be prioritized according to the risk evaluation in Table 8. The risk mitigations are ordered by main component and related to the identified risks by id in the following table.

Mitigation ID	Risk ID	Mitigation	Solved
1. Output platform			
1.1	1.1	The system shall provide secure communication mechanisms via SSL for processing user data regarding the login process.	YES
1.1	1.10	The system shall provide secure communication mechanisms via SSL for processing user data regarding the login process.	YES
1.2	1.2	The system shall not store the user password directly but in a hashed form.	YES
1.3	1.3	The system shall implement state of the art hashing mechanisms for sensitive user data.	YES
1.4	1.4	The system shall prevent brute force attacks on the login functionality by observing the amount of login tries as well as by slowing down the login speed (bcrypt).	YES
1.5	1.4	The system shall alert the administrator if a brute force attack is recognized	NO
1.6	1.5	The system shall enforce a strong security policy [1] regarding the strength of the user password.	YES
1.7	1.6	The system shall implement a secure password recovery mechanism if users forget their passwords.	YES
1.8	1.7	The system shall display only generic error messages during the login process to prevent information leakage.	YES
1.10	1.9	The system shall generate strong tokens for session management which cannot be guessed	YES
2. Profiler			

2.1	2.1	The system shall implement authentication mechanisms to prevent unauthorized access to the user profile.	YES
2.2	2.4	The system shall implement authorization mechanisms to prevent unauthorized users to access user profiles	YES
2.3	2.4	The system shall perform an input validation whenever an advisor enters new data into the user profile	YES
2.4	2.4	The system shall perform an input validation whenever data from other components imports data into the user profile.	YES
3. Rewarder			
3.1	3.1	The system has to guarantee that content queries by the rewarder component do not alter the data within the data store	YES
3.2	3.2	The system shall regard input from other components as untrusted and perform a validation on it before executing queries based on this input	YES
3.3	3.3	The system shall log all critical changes to the user profile to guarantee tractability of all queries.	YES
3.3	5.1	The system shall perform regular backups of the Content Repository	YES
4. Adviser			
4.1	3.3	The system shall log all changes in the user model during updates.	NO
4.2	4.1	The system shall perform regular backups of the user model.	YES
4.3	4.2	The system shall guarantee that components which only need read access to the data model are not allowed to alter any data.	YES
5. Content provider			
5.1	5.1	The system has to guarantee that only administrators are able to change (edit and add) content to the Content Repository.	YES
5.2	5.2	The system shall guarantee that users are only able to read content .	YES
5.3	5.3	The system has to log all interactions between users and the Content Repository	YES
5.4	5.4	The system has to restrict administrator access by the IP Address.	NO
6. Avatar			

6.1	6.1	The system is an Android Application, which sets up an application sandbox, which isolates the app data and code execution from other apps.	YES

Table 8 Risk mitigation strategies for identified threats

REFERENCES

[1] https://technet.microsoft.com/en-us/library/ff741764.aspx

[2] Writing Secure code (2nd Edition). Michael Howard, David LeBlanc. Microsoft Press, Redmond

[3] https://www.owasp.org/index.php/Threat_Risk_Modeling