HORIZON 2020
The EU Framework Programme for Research and Innovation

CAPABLE
CAncer PAtients Better Life Experience

Grant Agreement No. 875052
Start Date: 01/01/2020 (48 Months)

Deliverable No. 5.8

Refined Version of the Coaching System with All Interventions Required for the Two Considered Clinical Use Cases

Due Date: 31/10/2023
Submitted On: 30/11/2023

Coordinator	University of Pavia (UNIPV)
Deliverable Lead Partner	PUT
Contributing Partners	DEON, UoH, UNIPV, NKI
Contact	Prof. Silvana Quaglini
Email	silvana.quaglini@unipv.it
Website	www.capable-project.eu

Deliverable Type		
R	Document, report	
DEM	Demonstrator, pilot, prototype	**X**
DEC	Websites, patent fillings, videos etc.	
OTHER		
Dissemination Level		
PU	Public	**X**
CO	Confidential (Consortium members including the Commission Services)	
CI	Classified Information (Commission Decision 2015/444/EC)	

This deliverable is a part of a project receiving funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 875052.

Table of Contents

List of Figures

List of Tables

1. Versions History

Version	Date	Author	Comments
0.1	1/10/2023	SW	TOC and preliminary structure
0.2	13/10/2023	SW	Draft description of interventions introduced after D5.6
0.3	28/10/2023	SW	Extension of the textual content
0.4	8/11/2023	SW	Revision and further extension of the textual content
0.5	13/11/2023	SW	Updated presentation for the video
0.7	18/11/2023	SW	Prepared for internal review
0.8	20/11/2023	MP	Review
1.0	21/11/2023	SW	Updates according to review recommendations

2. Executive Summary

The aim of this deliverable is to demonstrate the final version of the Virtual Coach (VC) component including all clinical and non-clinical coaching interventions that are used at both sites – ICSM and NKI – participating in clinical tests of the CAPABLE system. We would like to note that VC provides coaching related to adverse events associated with therapies of various cancer types, thus it goes beyond the two initially assumed use cases of renal cell carcinoma (RCC) and melanoma.

The detailed presentation of VC at the *release candidate* (RC) stage, including the actor-model-based architecture and the two-level hierarchy of computer-interpretable guidelines (CIGs), was provided in D5.5 (Wilk et al., 2022), therefore for the sake of brevity in this deliverable we focus on new elements introduced in the last phase of the development and refer the reader to relevant parts in D5.5 if necessary. Specifically, these new elements include motivational and administrative workflows and rules (e.g., motivational messages for My Usual Walk capsule, rewards for performing capsules, psychological and nutritional support), as well as architectural components and external tools for improving the reliability of VC.

This demonstration comprises a textual description and a video. The text in this document summarizes the final state of VC and provides details on the new developments mentioned above. Selected – most relevant and interesting – new elements are then presented in detail and demonstrated in the video available at the following link: https://capable-project.eu/d5-8_demo/.

The structure of this document is as follows. In Section 3 we define the scope of demonstration. In Section 4 we provide an overview of the final version of VC considering its architecture, available CIGs and possible intervention and data enquiry types. In Section 5 we present the external monitoring and notification system that oversees VC in addition to its internal architectural components. Finally, we conclude with the summary in Section 6.

3. Scope of the Demonstration

The video that is part of this deliverable provides a more detailed presentation of selected developments introduced in the final version of VC. Specifically, it demonstrates:

- Refined actor model-based architecture with the System Watchdog actor and its operations.
- Overview of available CIGs and refined approaches to their modeling and execution.
- Overview of available intervention and enquiry types for defining action and data enquiry tasks in CIGs.
- Selected new CIGs that provide interventions not included in the previous deliverable or rely on new modeling or implementation techniques. Specifically, these are CIGs implementing blood pressure (BP) monitoring workflow, motivational support workflow and rules assigning rewards (milestones, badges and medals).

In D5.5 we used clinical scenarios and a dedicated simulator to present live operations of VC. However, given the diversified nature of the content included in this deliverable, it would be difficult to prepare a comprehensive and coherent clinical scenario. Therefore, operations of VC and generated interventions are illustrated with execution logs produced using a test instance of the CAPABLE system.

Since VC is part of the decision support backend, it does not have its own user interface (UI) and it communicates with the users through appropriate FHIR resources (e.g., `Communication` or `Observation`) that are exchanged with the Patient and Physician Apps (Dashboard) via Data Platform (DP). The frontend components were already presented in other deliverables, e.g., D4.4 (Gabetta et al., 2023) or D6.3 (Vasilyeva et al., 2022), therefore they are excluded from this document and we only show selected UI elements in the video to complement the execution logs.

4. Changes in VC and PROforma CIGs

VC provides active and comprehensive coaching for patients (and their home caregivers). It also offers limited support (e.g., notifications related to selected vital signs and completed questionnaires) for clinicians. VC and other backend decision support components constitute a hybrid execution environment for PROforma CIGs that allows for *specialized* CIG tasks that are executed internally by specific components and for *specialized* data items that are extracted from FHIR resources or established dynamically (i.e., computed or retrieved from provided data sources) by the components. Details are provided in D5.6 (Glasspool et al., 2022) and in (Kogan et al., 2023).

VC has been described in several previously published deliverables – in D2.2 (Peleg et al., 2021) we defined a set of requirements and preliminary architecture for this component, in D5.2 (Gilboa-Solomon et al., 2021) we presented the scheme of major VC operations, and finally in D5.5 (Wilk et al., 2022) we demonstrated the RC version of the component. Thus, to avoid repetitions, in this section we focus on summarizing the final status of VC and on presenting new architectural components and CIGs introduced after the publication of D5.5.

4.1. Refined Actor Model-based Architecture

VC is designed and implemented following the *actor model* – it is composed of multiple independent entities called *actors* (conceptually similar to agents in a multi-agent system) that work in parallel and interact by exchanging messages. In Figure 1 we present the high-level architecture of the final version of VC, and below we provide a brief description of employed actor classes (for more details see Section 4.1 in D5.5):

- *Case Manager Adapter* – interacts with Case Manager (CM) and notifies other actors about reported events.
- *Deontics Engine Adapter* – interacts with Denotics Engine (DE) to facilitate the execution of CIGs.
- *Data Platform Adapter* – interacts with DP to retrieve and store FHIR resources.
- *Coaching Controller* – manages the coaching process for all patients handled by VC, responds to CM events, and schedules execution of CIGs.
- *CIG Executor* – manages the execution of CIGs and communicates with DE via the Deontics Engine Adapter.
- *Task Executor* – manages the execution of a specific CIG task, communicates with DE via the Deontics Engine Adapter, interacts with DP via the Data Platform Adapter and invokes specialized CIGs.
- *System Watchdog* – identifies unhandled FHIR resources associated with events reported by CM and processes these resources by invoking properly parametrized master CIGs.

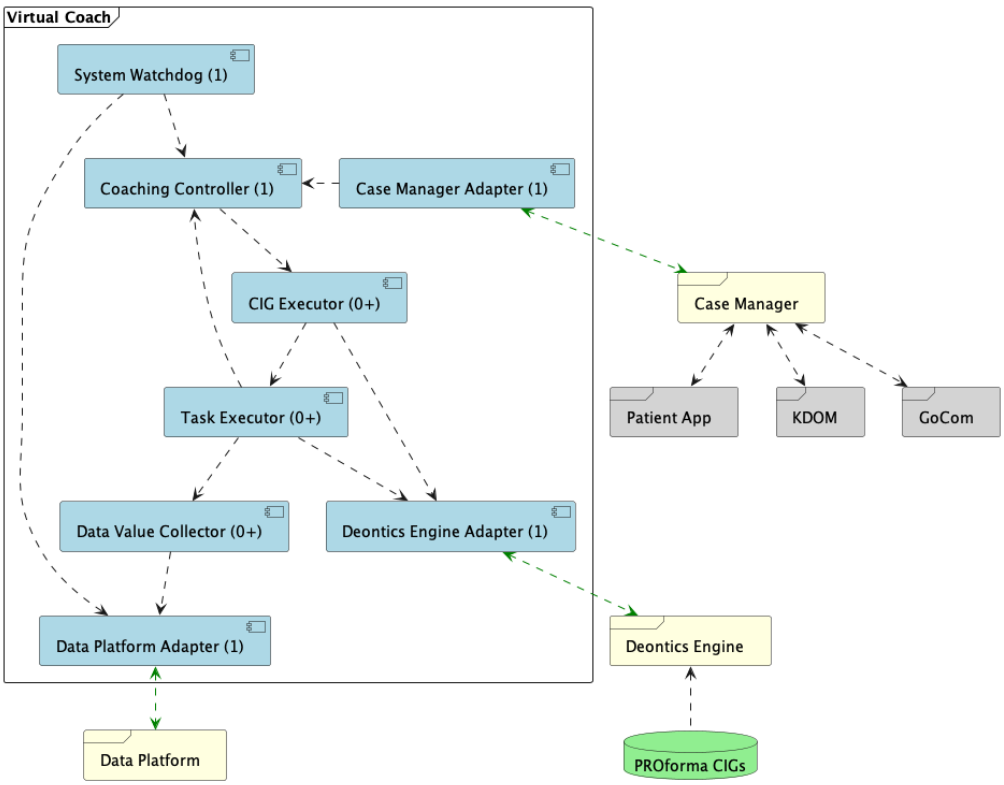

Figure 1. High-level architecture of VC (numbers in brackets indicate the number of specific actor instances)

System Watchdog was introduced in the final version of VC to deal with unhandled events that may be caused by internal problems in VC or by other components (in particular, DE that is critical for executing PROforma CIGs) being not available or reporting errors. While VC due to its *wait-and-retry* mechanism (i.e., in case of an error wait for some time and then re-retry – the wait time and the number of tries is configurable) can deal with a short-term unavailability of DE or other component (e.g., being stopped for several minutes for maintenance), longer absence periods require more complex counter measures realized by System Watchdog and described below.

In Figure 2 we present interactions between System Watchdog and other actors in VC – in the subsequent text we refer to specific interactions in this diagram. System Watchdog periodically (every hour by default) checks for unhandled resources. First, it retrieves recent resource *check-offs* [01] – auxiliary `Observation` resources indicating that a specific FHIR resource has been handled by VC (see Figure 3 for an example). The check-offs are created by a specialized task in the master CIG that is invoked for any event raised by CM (see Section 4.2 in D5.5 for details). Check-offs are stored in DP to ensure they are available after VC has been restarted – such an approach is also consistent with stateless approach to CIG execution adopted by VC.

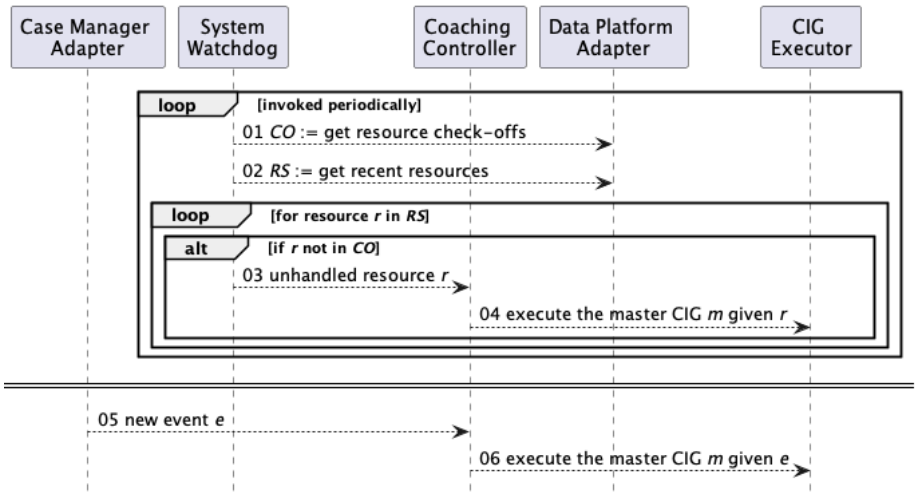

Figure 2. Flow diagram with interactions between actors in VC

Then, System Watchdog retrieves recent resources for which CM raised events and that should have been handled by VC [02]. Finally, System Watchdog iterates over retrieved resources and notifies Coaching Controller about resources with no check-offs [03]. In response, Coaching Controller executes the master CIG for a given unhandled resource [04]. By default, the time window for recent check-offs and resources is set to 24 h, and subsequent time window overlap – this allows us to successfully deal with situations when a problem persists for a longer period and is fixed only after several activations of System Watchdog, e.g., a component crashes at night and is restarted after several hours in the morning.

Figure 3. Example resource check-off and associated handled resource (an internal symptom reporting reminder)

The bottom part of Figure 2 illustrated how current events raised by CM are handled. The main difference is that for current events Coaching Controller is informed by Case Manager Adapter about an event (and an associated resource) to handle [05]. Then, execution of the master CIG for a given event is initiated [06]. The master CIG execution proceeds in the same way regardless of whether it has been invoked for an unhandled resource [04] or a current event [06]. A detailed diagram and description of interactions between VC actors is provided in Section 4.3 in D5.5.

In addition to introducing System Watchdog, we also revised the Case Manager Adapter actor to better handle errors that could occasionally occur when CM raises events associated with internal VC reminders (e.g., requests for symptom reports or questionnaires). Such errors were typically caused by temporary unavailability of DP and resulted in disabling a

corresponding CM rule, so subsequent events were ignored. Now Case Manager Adapter actively monitors and automatically recreates relevant CM rules to ensure the continuity of coaching services.

4.2. Available PROforma CIGs

In Table 1 we list and briefly describe all PROforma CIGs representing guidelines, workflows and rules that are used by the final version of VC. There are 31 CIGs in total – the master CIG and 30 specialized CIGs (see Section 4.2 in D5.5 for the description of the two-level CIG hierarchy). This relatively large number of specialized CIGs is associated with incremental introduction of the coaching functionality. Some of these CIGs could be combined as they are logically related (e.g., *vc-cig-new_symptom* and *vc-cig-symptom-report* or *vc-cig-new-visit* and *vc-cig-visit-reminder*), however, the authoring tool for PROforma (Composer) offers a very limited support for merging CIGs, therefore, we decided to keep them separately to avoid manual work and possible errors.

Selected CIGs are used only by specific institutions. For example, *vc-cig-bp-flow* – the workflow for monitoring and reporting BP – is used only in ICSM (NKI does not require patients to report their BP in the Patient App). Moreover, some CIGs are customized during run-time to the needs of a specific institution. For example, *vc-cig-study-start* and *vc-cig-study-end* that handle the start and end of the study for a given patient adjust a set of applied questionnaires – ICSM uses the BREQ2 (Behavioral Regulation in Exercise Questionnaire-2), while NKI does not. Consequently, *vc-cig-motivational-flow* for personalizing motivational tips related to the My Usual Walk capsule, in case of ICSM relies on BREQ2 responses to establish the order of the tips, while at NKI the order is fixed for all patients. Furthermore, capsule execution counts required to generate milestones and weekly rewards by *vc-cig-rewards* are different for NKI and ICSM.

Finally, some of the available CIGs have been excluded or disabled in clinical tests. Specifically, these are *vc-cig-nutritional-flow* and *vc-cig-sensor-check* for providing nutritional support and for checking the availability of sensor data, respectively. The former was excluded as the MST (Malnutrition Screening Tool) questionnaire was not fully implemented in the Patient App and ultimately the workflow was replaced by a periodic tip for the patient to report any nutritional problems during a follow-up visit. The latter was blocked due to reliability issues with the OmniCare platform managed by ASUS Life for collecting sensor data that were resulting in false alerts (i.e., a smartwatch is worn by the patient, but OmniCare fails to provide access to data) that could have been annoying and confusing for the patients and clinicians.

Table 1. CIGs available in VC (CIGs excluded from or disabled in clinical tests are marked with gra background)

#	Name	Version	Description
1	vc-cig-master	17	Master CIG
2	vc-cig-experience-sharing	1	Providing a recommendation to share experience after the symptom has finished
3	vc-cig-contact-level	2	Providing an appropriate contact level after a symptom has been reported

#	Name	Version	Description
4	vc-cig-immediate-feedback	3	Providing other immediate feedback (in addition to contact and guideline-specific recommendations) after a symptom has been reported
5	vc-cig-new-symptom	2	Setting a reporting reminder after a symptom has been reported
6	vc-cig-symptom-report	2	Handling the reminder to report a symptom and sending a reporting request if necessary
7	vc-cig-bp-flow	2	Managing a BP monitoring and reporting workflow – providing reminders for patients to measure BP and notifications for clinicians about abnormal BP patterns
8	vc-cig-insomnia	3	Monitoring insomnia using the Insomnia Severity Index (ISI) questionnaire
9	vc-cig-nutritional-flow	1	Managing nutritional support workflow and related questionnaires – see D7.1 (Sacchi et al., 2020) for details
10	vc-cig-psychological-flow	4	Managing psychological support workflow and related questionnaires – see D7.1 for details
11	vc-cig-form-fill-in	3	Handling reminders to complete questionnaires and sending corresponding requests to a patient if needed
12	vc-cig-my-usual-walk	1	Handling sensor readings related to daily steps and sending reminders to report the My Usual Walk capsule
13	vc-cig-other-capsule	1	Handling the report of other capsule than My Usual Walk and setting the reminders for prescribed capsules
14	vc-cig-rewards	4	Managing rewards (milestones, weekly and monthly badges, medals) for virtual capsule execution
15	vc-cig-motivational-flow	3	Personalizing motivational tips related to My Usual Walk based on patient behavior and evaluations of past tips
16	vc-cig-study-start	1	Handling the start of a study for a given patient and providing requests to complete relevant questionnaires
17	vc-cig-study-end	1	Handling the end of a study for a given patient and providing requests to complete relevant questionnaires
18	vc-cig-response	1	Handling the response to an intervention proposal provided by the patient though the Patient App
19	vc-cig-sensor-check	1	Checking the recency of sensor data retrieved from the OmniCare platform and notifying patients and physicians about potential problems
20	vc-cig-patient-tip	4	Providing motivational and preventive tips to patients based on their context (e.g., prescribed pharmacological treatment)
21	vc-cig-new-visit	1	Handling information about new visits and setting corresponding reminders
22	vc-cig-visit-reminder	2	Handling the reminder to notify about a visit and sending a corresponding notification to a patient
23	vc-cig-hobby-reminder	1	Handling reminders related to reported hobbies and sending encouraging messages to a patient

#	Name	Version	Description
24	vc-cig-medication-update	2	Handling new and updated medication prescriptions and sending notifications to a patient
25	vc-cig-rash-pruritus	2	Managing rash and pruritus according to the patient-oriented part of the ESMO guideline
26	vc-cig-paronychia	2	Managing paronychia according to the patient-oriented part of the ESMO guideline
27	vc-cig-alopecia	1	Managing alopecia according to the patient-oriented part of the ESMO guideline
28	vc-cig-mucositis	2	Managing mucositis according to the patient-oriented part of the ESMO guideline
29	vc-cig-hfs	1	Managing hand-foot syndrome according to the patient-oriented part of the ESMO guideline
30	vc-cig-fatigue	3	Managing fatigue according to the patient-oriented part of the ESMO guideline
31	vc-cig-diarrhea	3	Managing diarrhea according to the patient-oriented part of the ESMO guideline

Table 1 also specifies the version of each CIG that indicates the number of revisions. Most CIGs are at version 1—3 which indicates they have gone through up to 3 revisions and the most frequently revised CIG is the master one (version 17). Many of these revisions were associated with introducing new specific CIGs that are invoked by the master CIG in a specific context. Moreover, a significant restructuration was required for proper handling of various patient statuses during the study, like hospitalization, that affect the scope of provided coaching. For example, when the patient is hospitalized, they should not be reminded about symptom reports or should not receive motivational tips for My Usual Walk. The structure of the master CIG with sub-plans introduced to handle specific statuses in presented in Figure 4. The bottom part the figure also includes the *check_event_resource_off* task for marking a resource as handled that was discussed in Section 4.1.

Figure 4. Structure of the master CIG with sub-plans for handling specific events and various patient statuses

4.3. Refined Intervention and Enquiry Types

In Table 2 we present the *intervention types* that are supported by the final version of VC (from technical perspective they ascribe CIG action tasks). The entries in rows 1—13 indicate interventions already available in the RC version. Most of these interventions result in creating a `Communication` resource with a proper payload that is sent to the Patient App and then presented to the patient. The only exception is *cig-invocation* that invokes a specialized CIG from the master CIG. In the RC version this invocation was asynchronous, and now the execution mode has been changed to synchronous. The master CIG waits for completion of a specialized CIG in order to properly handle resource check-offs described in Section 4.1 – they are created only if event resources have been properly handled by appropriate CIGs.

Most new intervention types (rows 14—21) correspond to more complex operations that are performed by dedicated Task Executor actors. Tasks performed by these actors are invoked by specialized CIGs and then the control is passed from DE to VC. PROforma offers a limited support for programming constructs (e.g., for loops), or more complex operations on dates (e.g., find the date for the nearest Sunday), therefore more complex processing is "embedded" in VC. There are also simpler interventions, e.g., *message* that creates a `Communication` resource sent to the Physician App or *resource-check-off* (used by the *check_event_resource_off* task discussed previously) that creates an `Observation` pointing at a handled resource (see an example in Figure 3).

Table 2. Types of interventions supported by VC (interventions not used in clinical tests are marked with gray background)

#	Type	Description
1	medication-proposal	A proposal to start a medication or a set of medications
2	capsule-proposal	A proposal to start a well-being capsule
3	medication-update	A notification about a new or updated medication prescription
4	caregiver-contact	A recommendation to contact a health caregiver with additional instructions (so-called *contact level*)
5	prevention-tip	An educational tip related to symptom prevention, provided before the patient experiences a symptom
6	management-tip	An educational tip related to symptom management, provided after the patient has reported a symptom
7	infographic-tip	An educational tip pointing to infographic related to a specific treatment and symptom. Augments information provided in prevention and management tips
8	help-thermometer	A recommendation to use the "need help" thermometer for minor, but physically annoying, persistent symptoms and to contact a health caregiver
9	symptom-report	A request to provide an update on a symptom that has been already reported as ongoing
10	form-fill-in	A request to fill a questionnaire used to obtain additional information needed for proper patient management and evaluation
11	experience-sharing	A recommendation to share experience on the patient forum after a symptom has resolved
12	visit-reminder	A reminder about an upcoming (next 24 hours) appointment with the health caregiver
13	cig-invocation	A synchronous invocation of a specialized CIG
14	capsule-milestone-check	Checking if the patient is eligible for a milestone after performing a capsule and creating an `Observation` resource with the corresponding reward
15	weekly-reward-check	Checking if the patient is eligible for a weekly badge or medal for performing prescribed capsules and creating `Observation` resources with the corresponding rewards
16	monthly-reward-check	Checking if the patient is eligible for a monthly reward for performing prescribed capsules and creating an `Observation` resource with the corresponding reward
17	motivational-support-start	Initiating the motivational support for My Usual Walk, establishing the order of tips and setting relevant internal reminders (ticks) for VC
18	motivational-support-update	Updating the motivational support for My Usual Walk by changing the timing for sending tips and disabling tips discarded by the patient
19	tick-delay	Delaying an internal VC reminder (tick) when the patient is hospitalized, and an associated intervention is not applicable
20	resource-check-off	Creating an `Observation` with a resource check-off indicating that it has been properly handled by VC

#	Type	Description
21	message	A notification for clinicians and sent to the Physician App

Intervention types associated with rewards (rows 14—16) generate `Observation` resources that describe assigned rewards. Specifically, the structured description is stored as JSON serialized to string – see Figure 5 for an example. In this case we do not use `Communication` resources as for the majority of other interventions, as by design the content of the latter is represented as messages in the Inbox, and rewards are displayed in a dedicated section of the Patient App.


```
{
    "resourceType": "Observation",
    "id": "154803",
    "meta": {
        "lastUpdated": "2023-11-17T08:24:34",
        "source": "https://capable-project.eu/components/virtual_coach"
    },
    "status": "final",
    "code": {
        "coding": [
            {
                "system": "http://snomed.info/sct",
                "code": "228575009",
                "display": "Reward technique"
            }
        ]
    },
    "subject": {
        "reference": "Patient/828"
    },
    "effectiveDateTime": "2023-11-17T08:24:54",
    "valueString": "{\"text\":\"You did [[SCT:308931006|capsule-code]] [[5|string]] times. Great job!\",\"code\":
        {\"code\":\"308931006\",\"system\":\"http://snomed.info/sct\"},\"localizedText\":{\"arguments\":[{\"type\":\"capsule-code\",
        \"value\":\"SCT:308931006\"},{\"type\":\"string\",\"value\":\"5\"}],\"id\":\"capsule-milestone\"},\"rewardRank\":\"5\",
        \"rewardType\":\"capsule-milestone\"}"
}
```


Figure 5. Example of a reward (a milestone for performing My Usual Walk capsule 5 times)

We should note that two interventions – *infographic-tip* and *help-thermometer* – are not used by VC in clinical tests. Educational materials (including infographic tips) and the help thermometer form have been made freely available in the Patient App and the patient can access them any time whenever needed (this allowed us to limit the number of inbox messages presented to the patients).

In Table 3 we present the *enquiry types* that are available in the final version of VC (they characterize CIG enquiry tasks). Enquiry types in rows 1—6 were already available in the RC version. All new types (rows 7—12) are internal and they are handled by specialized Enquiry Task Executor actors. Most of these enquiries aim at retrieving various types of tips given the patient context (e.g., reported symptoms and prescribed treatments). The *internal-hobby* enquiry is different as it requires reading data from DP and then processing it. Specifically, it allows for selecting a random hobby (together with its additional description) declared by the patient in the latest response to the Activity Preference questionnaire represented as a `QuestionnaireResponse` resource. The selected hobby is then mentioned in an encouragement tip sent by *vc-cig-hobby-reminder* (see row 23 in Table 1).

The *internal-other-tip* allows for retrieving a localized (language-specific) versions of selected messages. Most of translations are handled by the Patient App, however, adding new translation resources requires updating the app which may be troublesome for patients.

Therefore, we introduced this enquiry type to handle tips introduced or updated after deploying the Patient App.

We should note that the *internal-capsule* enquiry in not used in clinical studies – automatic recommendation of virtual capsules has been finally replaced by a shared decision-making process where both the patient and the clinician choose capsules to be prescribed.

Table 3. Types of enquiries supported by VC (enquiries not used in clinical studies are marked with gray background)

#	Type	Description
1	external	Values should be retrieved from sources "external" to VC – DP or KDOM (detailed source is specified by meta-data associated with data items)
2	internal	Values should be generated internally by VC by running a personalized prediction model, performing some computations, or checking internal knowledge resources or repositories. This type is not used directly, but specialized to one of the following types given below
3	internal-capsules	Select capsules appropriate for selected clinical goals and hobbies indicated by the patient
4	internal-reporting-period	Select a reminder period for a given symptom and a reported description level.
5	internal-contact-level	Select a contact level appropriate for a reported symptom and its value (so-called description level)
6	internal-prevention-tip	Selected a random prevention tip relevant for the current therapy from a collection of tips based on the educational materials from AIMAC
7	internal-feedback-tip	Selects a proper feedback tip (e.g., a reminder to report additional symptoms) based on reported symptoms
8	internal-motivational-tip	Select a motivational tip related for My Usual Walk from a given category (attitude, capability and social support)
9	internal-capsule-frequency	Select a recommended weekly frequency for a given virtual capsule
10	internal-hobby	Select a random hobby from the set of pursued hobbies declared in the last response to Activity Preferences questionnaire
11	internal-bp-reminder	Select a proper tip related to BP monitoring
12	internal-other-tip	Select a localized version of a given message (if available in VC)

4.4. Refined CIG Execution Scheme

Initially, VC allowed for unrestricted parallel execution of CIGs (and their tasks). While this worked well during testing and under limited workload, concurrent executions of hundreds of CIGs resulted in DE running out of resources, in particular available connection sessions. Such situations were occurring when importing data from an electronic health record (EHR) system at NKI. During that time, hundreds of events (e.g., new prescription) were instantly raised by CM and VC tried to handle them by creating and running multiple instances of the mater CIG (one instance per event). To address this issue, we have enforced sequential execution of the master CIG for a given patient – specifically, at most one instance of the master CIG can be

executed for given patient at a given time and other instances wait in a queue. It is still possible to run multiple specialized CIGs for a given patient and to handle multiple patients at the same time. The new execution scheme is presented in Figure 6.

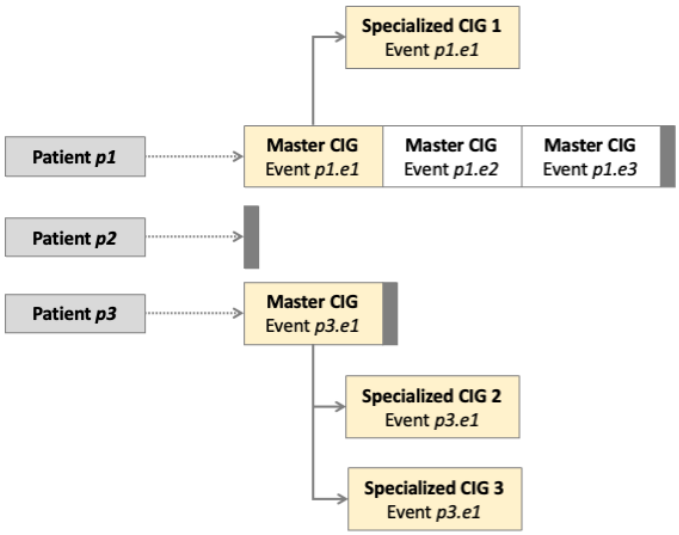

Figure 6. New CIG execution scheme (CIGs marked with yellow are executed in parallel, CIGs marked in white are waiting for execution, gray bar indicates the end of queue)

5. External Monitoring and Notification

Inspired by a monitoring and notification service developed for CM that periodically reports when specific components checked for the last time for their events - see D7.7 (Sacchi et al., 2023) for details - we developed a complementary solution for VC. The service is presented in Figure 7 – it relies on a Bash script and a Telegram bot. The script is invoked periodically by an operating system (1) (we use cron on Linux, the default frequency is every 2 hours) and it scans a daily VC log for error entries (2). If it finds any, it posts the 3 most recent entries via the Telegram bot (3), so that they can be immediately seen and checked by VC developers.

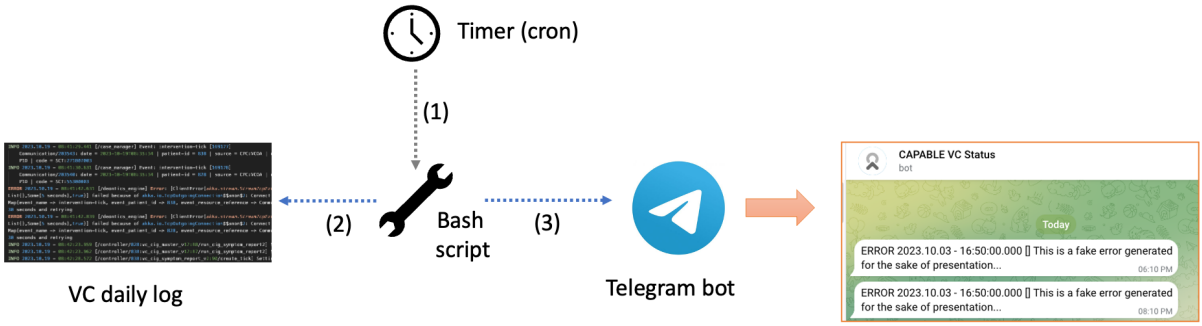

Figure 7. Monitoring and notification service for VC

While the CM-based service informs about major problems, such as a component being down or busy with activities other than checking for CM events, the VC-specific service allows for identifying errors that does not block VC from querying CM, but still may affect its operations related to coaching. For example, the service allowed us to identify and address issues such as non-responsive DE that ran out of disk space and had to be manually cleaned and restarted) or too strict timeouts DP-related operations defined in VC that had to be adjusted for increased system load and larger amounts of data).

6. Summary

In this deliverable, we have demonstrated the final version of VC used in clinical tests at NKI and ICSM focusing on the elements that were introduced after publishing D5.5 (the RC stage). Specifically, we have described the refined actor model-based architecture aimed at improving the reliability of the component, presented all available PROforma CIGs representing guidelines, workflows and rules, and discussed additional technical details related to execution of CIGs and their tasks (e.g., supported intervention types or enquiry types). Clinical and non-clinical coaching interventions offered by VC go beyond the two initially assumed use cases of melanoma and RCC and melanoma and can be used to help patients with other types of cancer, e.g., breast or lung.

We would like to note that some CIG and related functions (interventions, enquiries) were excluded from or disabled in clinical studies due to several reasons, with the most frequent one being the revision of requirements, however, they are still available in VC and can be activated as soon as they become needed.

In parallel with the refinement of VC, we have also conducted more theoretical work related to digital behavior change interventions (BCIs) and their personalization. In (Lisowska et al., 2023a) we have proposed a workflow for designing m-health BCI apps based on our experience with virtual capsules in the CAPABLE project. The workflow called SATO (IDEAS expanded with BCIO) expands the IDEAS framework with BCI customization and personalization steps and aligns them with the BCI Ontology. Moreover, in (Lisowska et al., 2023b) we have inroduced a BCI personalization framework based on the Fogg's behavior model (Fogg, 2019) that employs machine learning techniques (prediction and explanation with counterfactuals). The framework adopts a two-stage process where in the first stage we predict the "magnitude" of motivation, ability and trigger (MAT) factors in the Fogg's model and in the second stage based on these factors we predict whether the patient will adhere to a recommended BCI and engage in a desired behavior. If not, we first use counterfactuals to identify the MAT factors that should be revised improve the adherence, and then apply counterfactuals again to find detailed BCI characteristics aligned with the BCI ontology, e.g., schedule or dose, that should be revised to affect the identified MAT factor. Finally, we are currently exploring the possibility of using large language models to automatically generate personalized messages corresponding to various types of behavior change techniques.

We plan to continue this line of work after the CAPABLE project is finished, use SATO to design additional virtual capsules and integrate the personalization framework in VC.

7. Glossary

BCI	Behavior change intervention
BP	Blood pressure
CIG	Computer-interpretable Guideline
CM	Case Manager
CPG	Clinical Practice Guideline
DE	Deontics Engine
DP	Data Platform
RC	Release candidate
RCC	Renal cell carcinoma
VC	Virtual Coach

8. References

[1] Fogg, B.J. (2019): Tiny Habits: The Small Changes That Change Everything.

[2] Gabetta, M., Lanzola, G., Quaglini, S., Sacchi, L., & Wilk, S. (2023). CAPABLE D4.4: Final Platform Version and Deployment on all the Clinical Sites Involved. Zenodo. https://doi.org/10.5281/zenodo.7603442

[3] Gilboa-Solomon, F., Quaglini, S., Kogan, A., Glaser, S., Medlock, S., Barkan, E., Lisowska, A., Śniatała, K., Wilk, Sz., Tibollo, V., & Glasspool, D. (2021): CAPABLE D5.2: Framework Defined (Including Patients' Needs) Based on Available Data and Modelling Approaches, Zenodo. https://doi.org/10.5281/zenodo.5159285.

[4] Glasspool, D., Parimbelli, E., Quaglini, S., Kogan, A., Leizer, R., Barkan, E., Rabinovici-Cohen, S., & Wilk, S. (2022). CAPABLE D5.6: Prototype of Backend DSS, Ready for Integration with the Pilot System. Zenodo. https://doi.org/10.5281/zenodo.7603470

[5] Kogan, A., Leizer, R., Wilk, Sz., & Glasspool, D. (2023): A Hybrid Execution Environment for Computer-Interpretable Guidelines in PROforma. in: Parimbelli, E., Peleg, M. (eds.). Proceedings of KR4H2023: 13th International Workshop on Knowledge Representation for Health Care, pp. 82-89.

[6] Lisowska, A., Wilk, Sz., & Peleg, M. (2023a): SATO (IDEAS expAnded wiTh BCIO): workflow for designers of patient-centered mobile health behaviour change intervention applications. Journal of Biomedical Informatics 138, 104276.

[7] Lisowska, A., Wilk, Sz., & Peleg M. (2023b): Personalising Digital Health Behaviour Change Interventions using Machine Learning and Domain Knowledge. KR4HC 2023 – Knowledge Representation for Health Care. Workshop Proceedings, pp. 53-61.

[8] Peleg, M, Ganicheva, V., Lanzola, G., Panzarasa, S., Parimbelli, E., Polce, F., Quaglini, S., Sacchi, L., Veggiotti, N., Kogan, A., Leizer, R. Gabetta, M., Cornet, R., Glaser, S., Barkan, E., Gilboa-Solomon, F., Gisko, V., Śniatała, P., Śniatała, K., Wilk, Sz., Brunati, V., Ghio, V., Rizzo, M., Tibollo, V., Boekhout, A., Fraterman, I., Wilgenhof, S., Glasspool, D., Del Campo, L., Hernandez, L., Ottaviano, M., & Vicente, V. (2021): CAPABLE D2.2: Requirements Table and Use Case Description, Zenodo. https://doi.org/10.5281/zenodo.6058987

[9] Sacchi, L., Girani, E., Panzarasa, S., Parimbelli, E., Quaglini, S., Veggiotti, N., Peleg, M., Ghio, V., Rizzo, M., Tibollo, V., Boekhout, A., Fraterman, I., & Ottaviano, M. (2020). CAPABLE D7.1: Study Plan, Protocols Definition, and Informed Consent/Assent Drafts. Zenodo. https://doi.org/10.5281/zenodo.4540503

[10] Sacchi, L., Lanzola, G., Panzarasa, S., Quaglini, S., Veggiotti, N., Kogan, A., Leizer, R., Gabetta, M., Gisko, V., Śniatała, K., Wilk, S., Tibollo, V., Fraterman, I., Glasspool, D., & Ottaviano, M. (2023). CAPABLE D7.7: Interim Report on System Performances, Usage and Technical Improvements. Zenodo. https://doi.org/10.5281/zenodo.10144737

[11] Vasilyeva, A., Quaglini, S., & Peleg, M. (2022). CAPABLE D6.3: User Interfaces Prototype 2. Zenodo. https://doi.org/10.5281/zenodo.7096247

[12] Wilk, S., Quaglini, S., Veggiotti, N., Peleg, M., Glaser, S., Lisowska, A., Śniatała, K., & Glasspool, D. (2022). CAPABLE D5.5: Prototype of the Coaching System with Selected Representative Interventions. Zenodo. https://doi.org/10.5281/zenodo.7096208