Institute of Architecture of Application Systems

University of Stuttgart
Universitätsstraße 38
D–70569 Stuttgart

Masterarbeit Nr. 0838-011

Supporting Organizational Goals with Mobile Apps

Hanwen Cheng

Course of Study:	INFOTECH
Examiner:	Prof. Dr. Frank Leymann
Supervisor:	M.Sc. C. Timurhan Sungur
Commenced:	30. March 2016
Completed:	29. September 2016
CR-Classification:	H.4.1, H.5.3, J.7

Abstract

Informal Processes are human-centric business processes with unpredictable activities during modeling time. The Informal Process Essentials (IPE) approach aims for modeling informal processes in a resource-centric way so that the desired outcomes of informal process can be reproduced. Entities in IPE approach include organizational goals, capabilities, context, resources, and informal process. They are connected in organizations together.

Mobile application is widely used in common business to enhance business process from different aspects like flexibility and efficiency. Whereas support for informal processes using evolving mobile devices has not been investigated yet. In this thesis, requirements for organizational modeling will be investigated and satisfying solution concepts will be presented. As mobile applications have different use case scenarios and criteria compared with desktop applications, we derive the requirements from these use case scenarios and criteria. Hereafter, we propose new concepts, such as notifications and interactions, note list, privilege management, interactive group and interactive participants. To validate our concepts, we provide a case study on a mobile modeling scenario supported by an Android application implementing.

Our work introduces four added values compared current modeling approach for informal process, including time-involving definitions, location-independent modeling, interrelated participants and context aware definitions.

Acknowledgement

First of all, I would like to thank Prof. Dr. Frank Leymann and M.Sc. C. Timurhan Sungar for this great opportunity to let me work on this interesting topic. The frequent communication and discussion with Timur always give me stimulating ideas and systematic guide from both engineering and academic aspects. Furthermore, I would like to thank to Xinyue and my family for always supporting and encouraging me.

Contents

List of Figures

List of Tables

List of Listings

1 Introduction

A business process is a set of logically related tasks which is performed to achieve a defined business outcome [DS+90]. As modern business processes involve more and more intellectual workers, their forms are also changed gradually [OOM95]. People tend to present and discuss achievement in an electronic way regardless of where they are, equally work content could be easily modified in a collaborative way compared with pre-Internet eras [Car99]. As a consequence, the flexibility and diversity of work are increased, and the business activities become unpredictable, the business process involving such work and activities is recognized as *informal process* [SBD+16].

Business process modeling methods have been extensively researched by Jörg Becker et al. [BRV00]. In contrast, the modeling methods for informal processes still have huge potential. On the other hand, as mobile technology extends computing and the Internet into a wireless medium, and provides greater flexibility in communication, collaboration, and information sharing[WW05], mobile applications have significantly enhanced and extended the activities in business processes. In terms of these two aspects, the problem how to apply the informal process supporting model into a mobile scenario arises.

In this work, we will firstly introduce a mobile modeling scenario, then systematically study the requirement for mobile modeling based on a criteria catalog and informal process supporting model, and present a solution to create new concepts to supporting mobile modeling, after that we will dive into a case study and provide added value based on the implementation and validation of these concepts.

Overview

This thesis will focus on the modeling method to support achieving organizational goals in informal process within a mobile application context. Specifically, it is organized in the following manner.

1 – Introduction: This part offers a brief introduction about the purpose of thesis.

2 – Background: This part provides the basic definitions, fundamentals, technical contexts and a motivation scenario to understand the work.

3 – Requirement Analysis of Supporting Model: This part combines the analysis of mobile technology criteria with informal process supporting model, accordingly derive the requirements for supporting a mobile modeling scenario.

4 – Mobile Modeling Concept This part elaborates the concepts to realize aforementioned requirements, which includes detail the extended definitions and mobile modeling supporting features to be implemented.

5 – Mobile Modeling Case Study: This part discusses the necessary details of the mobile modeling application, utilizes a use case to validate the concepts, and finally concludes the added values in our application.

6 – Conclusion and Future Work: The final chapter summarizes the work has done in the thesis, and proposes suggestions for the future work.

2 Background

In this chapter, all the background information is provided. In the following we will mostly state these two parts in parallel, because the main focus of this thesis is intersection of organizational goals and mobile applications.

2.1 Fundamentals

In this section, we provide some fundamental definitions and information required to understand organizational goals. As a set of logically related tasks [DS+90], **business processes** are the key instruments for organizing business activities and for improving the understanding of their interrelationships in our society[DMR15]. A **business process model** is used in many organizations to capture recurring activities and the structures (control flows, data flows, etc.) of business processes [SBBL14], they also depict how various tasks are coordinated to achieve specific organizational goals [MDS+10].

Knowledge-intensive Processes (KiPs) are the activities whose conduction and execution heavily depend on knowledge workers performing various interconnected knowledge-intensive decision which create tasks [DMR15]. KiPs are genuinely knowledge-, information- and data-centric, moreover they require substantial flexibility at design and run-time [VHH+11]. Due to this reason, standard business process modeling method [BRV00] is not suitable for such processes. These processes are called **informal processes** on account of the involved unpredictable activities [SBD+16].

Sungur et al. have proposed a resource-driven approach called Informal Process Essentials [SBBL14]. In contrast to the unpredictable activities in knowledge-intensive process, goals of informal processes are known before the enactment [SBLW15][DMR15]. The goal could be modified indeed, but editing the goal could rather be a trivial work compared with activity-oriented models [SBBL14].The approach enables a certain degree of automation by engaging the required interrelated resources towards **organizational goals** automatically [SBD+16]. The organizational goal refers the constraints, or sets of constraints, imposed by the organizational role [Sim64], it will be referred as organizational intention as well in our following work.

2.2 Motivation

In this part, we will give two motivation scenarios and illustrate the reason why we focus on this topic.

2.2.1 Informal Processes

As our technology advances and the globalization increases, the most valuable asset of a 21st-century institution (whether business or non-business) will be its knowledge workers and their productivity [Dru99]. These knowledge workers tend to work in a collaborative and ad-hoc way, which connect frequently with external resources and hard to predict. In terms of the related resources, it not only refers to the means of production, but more importantly, human-centered knowledge resources, i.e., the intellectual resources.

Informal processes involve knowledge workers in decision-making processes, development processes and maintenance processes. They play a critical role for many organizations because they represent a valuable amount of the work among what organizations deliver. Therefore, supporting the modeling of informal processes is vitally important for organizations [SBLW15]. Moreover, since the operations and strategies for a company are frequently adjusted and unpredictable, they will have variant means to achieve an organizational goal, and it is hard to determine the definitions and modeling business process. Whereas it is sufficient to use aforementioned IPE approach to model the business processes based on their resource without necessarily specify their activities [SBD+16].

The organizational informal process focuses on such dilemma and aims for overcoming the mentioned obstacles.

2.2.2 Development of Mobile Applications

In 2009, worldwide mobile application downloads in leading app stores amounted to approximately 2.52 billions and are expected to reach 268.69 billions in 2017 [1]. Furthermore, owing to the continuous development of mobile technology, mobile apps are capable of anything [Cle11], and now they substantially contribute to a collaborative team.

[1] http://www.statista.com/statistics/266488/forecast-of-mobile-app-downloads

On the one hand hardware evolves a lot, e.g., CPU clock rate, memory size, GPU computing capability, complete integrated sensors and brand new chips like Bluetooth Low Energy (BLE) and Near Field Communication (NFC). On the other hand software platforms ceaselessly iterative and incrementally develop, which include the compiling algorithms, the open API (Application Programming Interface), new mechanism like push notification [BBD11], the appearance of new mobile application framework like Ionic [2] and React Native [3]. All these mentioned aspects dramatically make the mobile engineering smoother and provide more functional possibilities for applications. As more functions are enabled, and enhanced performance is integrated into mobile devices, we use mobile applications frequently to replace other instruments. For example, a chip card for house controlling, a photo for instant information recording, a video meeting connector, and even in advanced project management. According to Hoos et al. recent process-driven analysis [HGM15], engineering apps can significantly enhance flexibility and efficiency of business processes. By this reason, it is urgently needed to extend the support of existing project to a mobile scenario.

2.2.3 Motivation Scenario

BlockTech is a web technology company based in Berlin, with developers team settled in Stuttgart, aiming for creating an Internet financial technology (FinTech) product, they have just set up the team and completed a business plan. As a starting company, they have to frequently discuss about the further strategy and consult external experts in both business and technical fields. Therefore, their business process is highly unpredictable at the moment. In addition, they have a distributed team, and team members have frequently business trips.

Mobile application is widely used in common business, for example, many collaborative teams highly depend on business process management software like Wrike [4] and Redmine [5] for coordination and communication. Executive officers in BlockTech are considering using mobile applications to enhanced flexibility and efficiency of normal business process [HGM15] and overcome some obstacles like geographical dispersion [Car99]. But the business process modeling on these applications need specifying activities, which is not possible in unpredictable business process of BlockTech. They are looking for a way to address both the informal process and the mobility of participants.

[2] http://ionicframework.com/
[3] https://facebook.github.io/react-native/
[4] https://www.wrike.com/
[5] http://www.redmine.org/

Inspired by this **mobile modeling scenario**, in this thesis we will create concepts for informal process and implement the concepts to support organizational goals. In the Chapter 5 we will give a detail case study based on the mobile modeling scenario, furthermore, we will introduce added value regarding to concepts.

2.3 Related Work

In this section, we will detail the previous scientific research about informal process. Besides, we will review the previous study on mobile application design. The implementation of related work will also be presented here.

2.3.1 Informal Process Essentials

Sungur et al. have proposed a model called Informal Process Support Model (IPSM). It consists of Informal Process Essentials (IPE) which we mentioned before and Informal Process Recommendations (IPR) which are the guidances and tips gathered from previous executions.[SKL14]. More specifically, IPE approach enables describing and enacting the informal process based on existing resources and available knowledge of human performers, which will be detailed in the following paragraph. The IPR declares a way to smooth the execution by using the previous knowledge as recommendations. The key is such recommendations will not constrain the flexibility of whole model, they are based on existing or previous enactment.

Informal Process Essentials detailed the resources to create executable informal process models [SBLW15]. It models the existing resources and knowledge for supporting the process enactment. The definition of models, the interrelationship between resources and the organizational goals are created by experienced performers, after enactment, the converted instance will enable processing by human performers in an autonomous way.

As a resource-centric approach [NEK+05], Informal Process Essentials enables a certain degree of automation by engaging the required interrelated resources towards *organizational intentions*, and it also documents the *resources* like data resources and tools required during process execution [SBD+16]. Therefore, organizational intention is a core entity in our models, both collaborations and processes aim for achieving an intention. *Resources* are needed for corresponding intention, and executives will make strategic decision based on the provided resources [SBBL14], in our case, these decisions will be referred as *organizational strategies*.

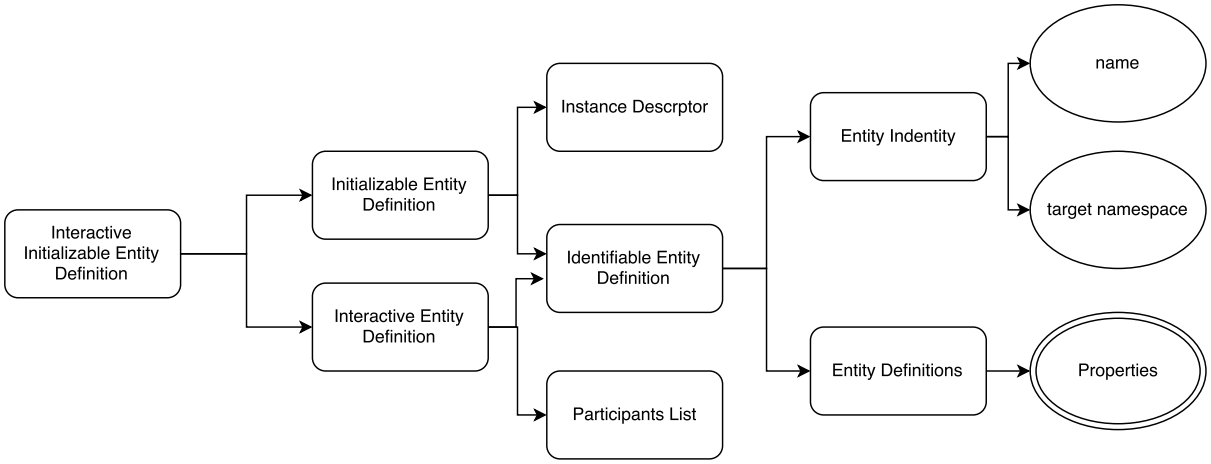


Figure 2.1: Entity Diagram of Informal Process Essentials

The relationship of entities could be shown in Fig. 2.1. Specifically, intentions, strategies, contexts, capabilities, informal processes are the *Identifiable entity definitions*. They all have *Entity Definition* and *Entity Identity*, where identity denotes to the name and target namespace, and definition uses variant properties to distinguish different models. Moreover, These definitions all have a *Participants List*, and administrator could have control privilege according to this list, they are actually a group of *Interactive Entity Definition*.

Intentions, strategies, capabilities and informal process could have variant instance models based on their definition, they are categorized as *Interactive Initializable Entity Definition* in IPSM, and instance will be specified by *Instance Descriptor*. Context highly depends on the time and circumstance, thus context does not have concrete instance. And resources are the representations of available organizational resources [SBD+16], it does not have instance either.

Business experts could instantiate definitions as a new instance under the condition that they fulfill all conversion requirements. And conversely, a definition could also be extracted from an instance in order to create further similar instances, thus create new binding with current instance descriptor and derived definition [Kal16] [SBBL14].

Instance descriptor to assign the detail attributes for each instance. Descriptors of different definitions share a same schema with similar elements, in other words, intention instance, strategy instance, etc., share a same structure. The properties of an instance are:

- State
- Parent Instance

- Instance URI
- Source Model
- Start and Due Date

Entity Definitions

According to [SBBL14] [Kal16], the following list presents some brief introductions about these definitions, and Fig. 2.2 presents their relationships.

Intentions depict what to be achieved in an informal process, the intention will guide the whole process during modeling and execution, each process should accordingly target at an intention, e.g., *increasing revenue*, *setting up a team*. Additionally, the intention could have related intentions with different levels of detail. Intentions includes following properties:

- Initial Context
- Final Context
- Related Intentions
- Achieving Strategies
- Priority

Strategies explain abstractly the countermeasures to achieve an intention. Compared with abstract goal, it is rather a concrete solution, e.g., *recruiting more employee*, *meeting with partner company*. Strategies include following properties:

- Target Intentions
- Required Intentions
- Operational Processes

Informal Processes are the core in informal process supporting model. Informal processes integrate all the related entity definitions, aim for implementing an strategy, target at certain goal, and include available resources. They includes following properties:

- Source Model
- Target Intention
- Required Intentions
- Operational Capabilities

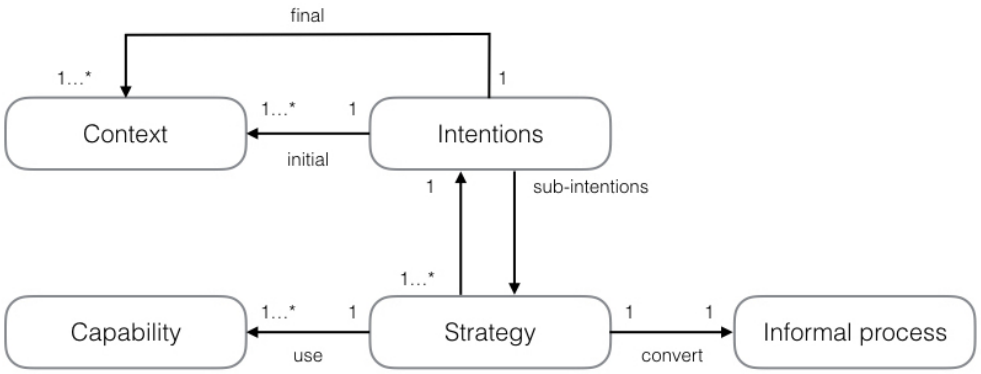


Figure 2.2: IPE Basisc Meta Model

- Resource Model

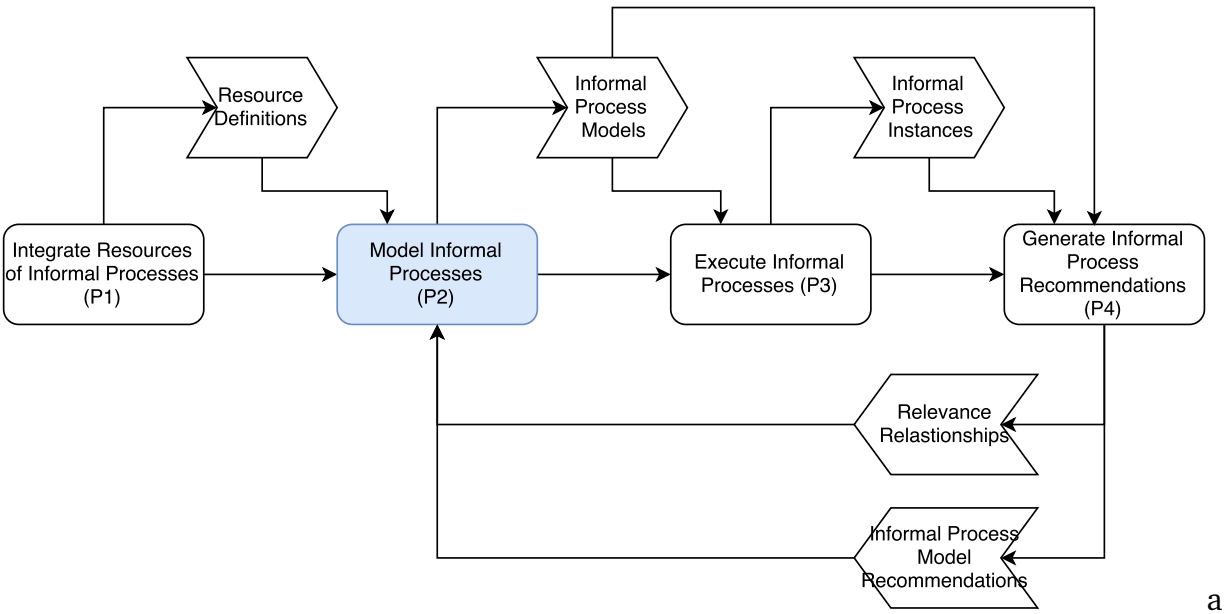
Capabilities represent abilities to perform tasks in informal processes. They enable a resource-centric way to express an informal process. Capabilities are not limited to real properties, they could also refer to a group of technical experts who have certain capability, the basic infrastructure or utility, and even the abstract relationship. Capabilities includes following properties:

- Type
- Contained Capabilities
- Providing Resources
- Desired Resources

Contexts are the related circumstance of processes. They could be specified in variant aspects according to intentions,. For example, the number of employees, the amount of profit, and information collected by Internet of Things (IoT) technology [AIM10] are contexts. It is not only the current problem description, but could also be the qualitative or quantitative specification of an abstract goal. It is always judged with an intention as a supplement.

Resources are the concrete elements to realize capabilities. Compared with capabilities, resource is a single instance with all its component and information, rather than the description of a common capability among several instances.

The IPSM [SKL14] is a basic framework for informal process to be carried out, and the Informal Process Essentials is a resource-driven business process modeling and execution approach, from where we will derive our support models. In the rest of this thesis, **definitions** will refer to identifiable entity definitions, and **entities** will refer to instance descriptors, participants lists and identifiable entity definitions.



a

Figure 2.3: Role of informal process modeling in InProXec method

2.3.2 InProXec Method

To understand our focus in this work, it is better to have a general view of the informal process model in a broader context. The approach to automate the enactment of informal processes could be illustrated in Fig. 2.3. The execution of informal processes could be detailed within four steps, referred as InProXec method[SBLW15] [SBD+16].

In InProXec, modeling is the second step. Before modeling an *Integrate Resources of Informal Process* is needed, thus the resources from different origin and forms could be allocated in a uniform way. After modeling, the model will be improved by actual functionalities, the execution on model will be enabled. This phrase is called *Execute Informal Process*. Thereafter, a final step called *Generate Informal Process Recommendations* take place, where the relevant relationships and corresponding informal process model recommendations are presented.

The modeling phrase (P2) uses the uniformed resources from first step as input in P1, then create an organizational goals with related elements, and finally outputs a model with declared functionalities to be researched in P3. It has close relationship with methods and platforms in other phrases. So a use case study in the whole could clarify the ideas and concepts of supporting and executing informal processes.

Sungar et al. proposed a supporting infrastructure based on existing cloud-computing technologies [SBBL14]. It proposed the a Informal Process Support Center (Co Act), which is under development, mobile application **Co Act Mobile** will take place in this

part to support modeling organization goals, and we will validate our concept and implementation in this part. The whole informal process support center will establish a complete support solution to connect user, IPE repository and resource organizer. To be more specific, the resource organizer provide a complete interface to integrated resources, map the hADL [DT12] elements and their relations, prepare all information required for deploying and initializing resources upon informal process [SDDL15]. Resource organizer uses Winery API to fetch available resources [KBBL13] and openTOSCA to initialize the corresponding resources in the cloud infrastructure [BBH+13].

With the understanding of related work and whole architecture in our studied case, it is also clear that what process we will interact with, and what infrastructures we could utilize in our work (AKA. resource organizer and IPE repository). In these two dimensions our role is clear in supporting informal processes.

2.3.3 Mobile Apps in Business

Since the iPhone AppStore opened in July, 2008[6], there is an exponential growth in mobile apps industry [Cle11], and mobile application engineering has become an emerging research area [Was10]. In the following, we will introduce mobile application specific features, design methods, and the added-value from a business point of view.

Specific Features

Mobile app has its own limitation due to its screen size, its computing capability, etc., whereas it also has remarkable advantages compared with desktop applications. For instance, it integrates plenty of sensors, it can detect gestures, collect location information to extend its functionality and improve user experience in order to achieve a context-aware functionality[LKAA96]. Moreover, it is always with us.

Notifications are used to notify a person or a group of people with a noteworthy business event, such as that a particular order has been approved, or a particular product is about to be shipped [AAD+07]. Push notification describe an Internet request initiated by a “publisher” or server [LO12]. When participants collaboratively work together, it would be common if that they want to leave some messages regarding to some issues in processes [AAD+07]. A well-designed notification subscribe mechanism will broadcast necessary information. Push notification function is integrated smoothly into modern mobile platform, which could improve the efficiency of communication.

[6]http://www.macworld.com/article/1134380/app_store.html

Design Method

According to Wasserman's research [Was10], mobile applications present some additional requirements which are less commonly found with traditional software applications. These requirements, in return, have a strong influence on application development. In general, there are four aspects to be considered:

- User experience
- Non-functional requirements (e.g., network)
- Processes , Tools, and Architecture
- Portability

Harrison et al. [HFD13] have emphasis the importance of cognitive overload in their People At the Center of Mobile Application Development (PACMAD) usability model. Cognitive load refers to the amount of cognitive processing required by user for interacting the application [Swe88]. In a mobile context, cognitive overload is a regular situation, and mobile devices may be particularly sensitive to the effects of cognitive overload, that means, user can not focus only in one application, the dispersion and complex work in business process and common life will overload user's dispersion. Due to the reason informal process has a more flexible and unpredictable circumstance [SBD+16], it is more frequent to face such a problem in mobile modeling. An experienced collaborator need to make modeling decision in ad-hoc way, then team member will update the model during their work. For example, a manager will edit the process strategy definitions when he waits the train. Therefore, we will consider intuitiveness which reduces cognitive loads as a design guidance in our following work.

Sandhu et al. [SCFY96] has prompt an efficient role management called role based access control (RBAC) model, The roles have a hierarchical relationship rather than existing independently, a high level role will include all the right in lower level roles.

Eva Hoos [Hoo14] has proposed a design method for Mobile Engineering-Application Middleware (MEAM). The MEAM method aims for facilitating and standardizing the realization of mobile apps in engineering and fitting in the existing IT environment. It mainly includes four steps:

- Mobile app usage characterization
- Use case identification
- Requirements analysis
- Concept development

We will elaborate the first three steps with our use case models in Chapter 3 and resolve a complete concept in Chapter 4

Value-added Application

Mobile apps used in business are called mobile enterprise apps [GSD12]. They can optimize or extend business processes and enhance employee's productivity [HGKM14a]. The added value need to be analyzed before the implementation. With qualitative evaluation based on comparison, Eva Hoos et al. [HGKM14a] have presented a holistic analysis framework for the goal-oriented use of mobile technology in business processes to identify value-added usage scenarios of mobile technology with a special focus on mobile apps. Based on this framework, a comprehensive analysis method to identify such apps is presented, which is called ValueApping. ValueApping detailed the method in four steps, namely:

- Process analysis
- App potential evaluation
- Recommendation generation
- Business benefit analysis

Within these steps, it utilizes multi-criteria analysis, portfolio analysis and metric analysis techniques in requirement analysis [HGKM14b]. We incorporate such an approach for requirement quantification and comparison, which will be detailed in Section 3.2.

Understanding the mobile engineering with the improvement of mobile technology involves many complex issues, so in this work we are not able to focus on the common mobile software design and engineering method, but we try to collect comprehensive scientific knowledge for organizational goal modeling.

With the holistic view on the top of essentials, mobile apps, and related works, we will analysis the requirement in the next chapter.

3 Requirement Analysis of Supporting Model

In this chapter, we will analysis the requirement of our topic. Requirements for mobile modeling relate not only to the application mobile potential, but also relate to informal process models. In order to get the hybrid requirements from these two fields, we need to adopt a scientific analysis method in order to get the conclusion.

3.1 Criteria Catalog

Hoos [HGKM14b] has prompted a method to define the potential value of mobile applications based on a criteria catalog. Such catalog has covered many aspects in business activities, and has stated some fundamental criteria for mobile device industry. In this part we will extend the criteria catalog and give the definitions of each criterion in informal processes. Compared with Hoos's work, we introduce new criteria like synchronization frequency, correlation, acknowledgment, fault tolerance and consistency compared with Hoos's criteria catalog [HGKM14b] as shown in Fig. 3.1.

Participant Mobility (C1) involves how frequently a participant in informal process is not in his office or work station. The scenario where the operator uses laptop is

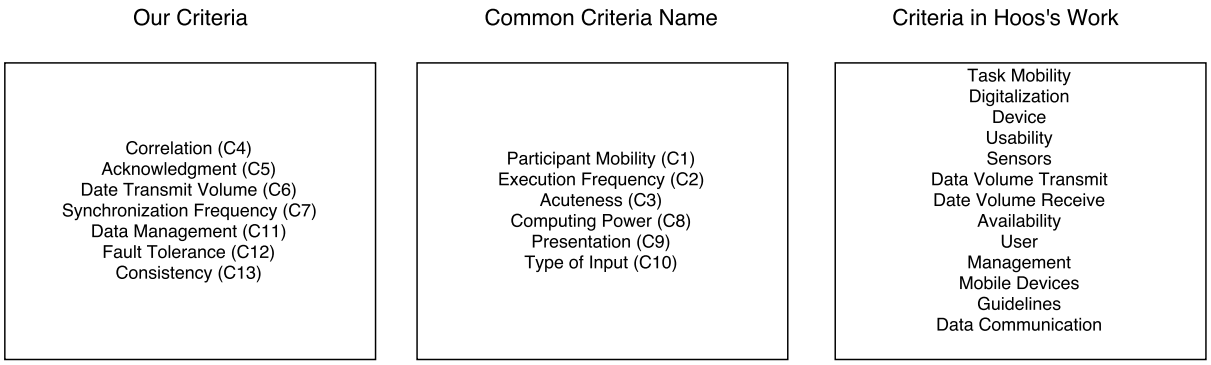


Figure 3.1: Criteria Namespace Comparison

not taken into account. The mobility refers particularly to situations where the operator cannot use the traditional application, e.g., desktop application. We define our supporting model in a high mobility circumstance across all the definitions.

Execution Frequncy (C2) for operation is how often the corresponding definition need to be operated or modified in an informal process. It is the amount of execution needed on mobile modeling in each step. The frequency depends on the number of tasks needed to be performed, and the number of interactions needed for each task. Resource definitions are created and allocated during the step of *Integrating Resources of Informal Processes* (see Section 2.3.2 on page 24). Capabilities are also predefined before making the strategic decision, so they have a regular execution frequency. A contexts is rarely updated after creation since it records the state at a certain point of time. In contrast, intentions, strategies, and informal processes need to be adjusted frequently according to organizational strategic decisions.

Acuteness (C3) describes every definition in time dimension, it depends on how quickly user need to respond the process. For example, the definition which needs to take instant reactions will have a high acuteness. In our supporting model, resources and capabilities are predefined, in contrast strategy and informal process may directly affect the running instance, so they have a high acuteness.

Correlation (C4) denotes the relationship between different definitions. This is a specific criterion in our informal process since we have variant definitions. The correlation will affect the structure of entities and design of mobile modeling. According to description in Section 2.3.1 on page 22, strategies, intentions and informal processes have high correlations with other definitions.

Acknowledgment (C5) is the importance of sharing the information of corresponding definition among participants. A collaborative team needs to synchronize team members' work according to different steps in an informal process. Participants also need to adjust workload or their focus depending on other's progress. For example, a back-end developer has to adjust the handler function of API endpoint regarding to data specialist's work on database. It is important to broadcast modifications of definitions to all participants in terms of acknowledgment.

Date Transmit Volume (C6) refers to the amount of data transmitted in informal process. It related to both of receiving volume, and sending volume. Though modern applications mostly adopt asynchronous request, users still need to wait for a successful acknowledgment from server in order to continue further executions, especially in the condition which is below average Internet connection quality. As resource definitions are the representation of organizational resources available in modeling and execution [SBD+16], they may relate to extra external resources,

which leads to a large data transmit volume. In contrast, the data transmitted in other definitions, is mainly references and textual information.

Synchronization Frequency (C7) refers to the oftenness that a synchronization is needed between client and server. The frequency of each definition depends on how often a definition is updated by other participants. In our support model, resources will be frequently updated externally, and informal processes need to be frequently modified in order to fit running instances.

Computing Power (C8) related to the calculation capability needs to process in certain definition. As modern mobile phones ceaselessly improve their CPU computing power, normal computing task will rarely cost long execution time in an informal process. *Initializable Entity Definitions* need normal computing power for creating instances and extracting definitions. Particularly, strategies can be transformed to informal processes, during the conversion extra computation related to other definitions is introduced. Therefore, strategies need a strong computing power.

Presentation (C9) counts the information in each definition to be presented with the display limitation of the mobile phone. Even though by minimizing the font size, more information could be displayed, whereas a heavy cognitive load (see Section 2.3.3 on page 26) is introduced at the same time since more information occurs at once. Therefore, these criteria decide the display structure and styles of a definition. We count and compare all the attributes and element included in each definition according to description in Section 2.3.1 on page 22. Strategies, intentions and informal processes have the most attributes, also resources could be related to organizational resources, thus they have much information to be presented.

Type of Input (C10) is the complexity of user input. Mobile phones integrate a lot of sensors to give different real-time information like location, accelerations, etc., plus embedded cameras can capture photo or QR code as input information. Here the input criterion counts the demand of a complexity input data in mobile modeling. As capabilities describe the ability, and resources have their own structured representation, no complicated input is required. In contrast, other definitions require complex input as contextual information.

Data Management (C11) evaluates the amount and structure of data to be stored into mobile application according to different definitions. In terms of the increasing storage in mobile phones, criterion is not highlighted as before. For instance, the basic model of iPhone 6s is equipped with 16 GB storage which is able to maintain about 5000 high-quality images. But the efficiency in terms of retrieving and reforming data is still a crucial point in each definition.

Fault Tolerance (C12) is the capability that an application can handle the incorrect processes. It also reflects cost to fix these errors. The errors are not limited to the external Internet error, they can also refer to the human-fault activities. A basic requirement is that a failure operation or Internet error should not stop the execution of application. In our case, *Initializable Entity Definitions* need a high fault tolerance as these definitions have complex relationships with instance descriptors.

Consistency (C13) considered with functionality and the user interface (UI) design. A consistent structure could reduce the cognitive load for actor, smooth the user experience, furthermore improve the process efficiency. It is highly required in all our definitions.

Building mobile application involves works in many fields, eg., software engineering, management, design, service quality etc. In order to get the criterion which has the greatest impact on our informal process supporting model, a criteria catalog will be used as a basis.

In this section, we have listed our specific criteria especially focus on the mobile modeling scenario. In next section, we will use definitions in the supporting model to evaluate each criterion.

3.2 Evaluation Method

In order to get the weight of criteria based on the mobile modeling scenario, firstly, the influence on each definitions in informal process will be judged according our previous introduction of criteria. Consistently, we will use three levels to indicate the influence for all the process, which are illustrated in Table 3.1.

The next step is to quantize and summarize the influence of different definitions on criteria. We will use a concrete numeric value $s(k_{c,d})$ to indicate the influence for certain criterion c and certain definitions d, where k_c denotes to the ordinal value for certain criterion. We calculate the final score by the formula Eq. (3.1).

$$k_c = \sum_p s(k_{c,p}) \tag{3.1}$$

These values start from 0 instead of 1, which is easy for us to set the threshold to compare the importance, hence the relatively less required criterion will range from score 0 to 6, accordingly, the criterion with high demand range the score from 6 to 12.

Table 3.1: Criterion Value Quantization

Score	2	1	0
Participant Mobility (C1)	High	Medium	Low
Execution Frequency (C2)	Frequently	Regularly	Rarely
Acuteness (C3)	High	Medium	Low
Correlation (C4)	High	Medium	Low
Acknowledgment (C5)	Important	Normal	Trivial
Data Transmit Volume (C6)	Large	Medium	Small
Synchronization Frequency (C7)	Frequently	Regularly	Rarely
Computing Power (C8)	Strong	Normal	Weak
Presentation (C9)	Much	Medium	Few
Type of Input (C10)	Complex	Normal	Simple
Data Management (C11)	Complex	Normal	Simple
Fault Tolerance (C12)	High	Medium	Low
Consistency (C13)	High	Medium	Low

Table 3.2: Criterion Evaluation

Score	Intention	Strategy	Context	Capability	Resource	Informal Process	Total
Participant Mobility (C1)	2	2	2	2	2	2	12
Execution Frequency (C2)	2	2	0	1	1	2	8
Acuteness (C3)	1	2	1	0	0	2	6
Correlation (C4)	2	2	1	1	1	2	9
Acknowledgment (C5)	2	2	2	2	2	2	12
Data Transmit Volume (C6)	1	1	1	1	2	1	7
Synchronization Frequency (C7)	1	1	1	1	2	2	8
Computing Power (C8)	1	2	0	1	0	1	5
Presentation (C9)	2	2	1	1	2	2	10
Type of Input (C10)	2	2	2	1	1	2	10
Data Management (C11)	1	1	1	1	1	1	6
Fault Tolerance (C12)	2	2	1	2	1	2	10
Consistency (C13)	2	2	2	2	2	2	12

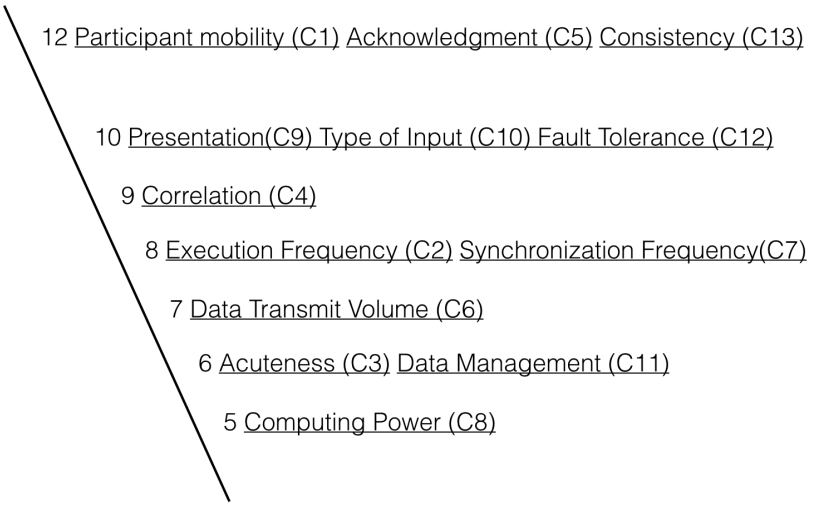


Figure 3.2: Criteria Order

3.3 Concluded Requirements

According to our analysis, we have the following priority order Fig. 3.2 based on analyzed criteria's importance regarding to the mobile modeling scenario. In addition, from different aspect of view, we will conclude the following requirements of modeling on a mobile scenario.

Collaborative teamwork (R1): In a cooperative point of view, we consider the criterion C1, C3, C7 and C13, each informal process involves several participants. Thus a collaborative teamwork environment is the prerequisites to start a process, the questions like who starts the process, who sets the goal will be asked by the participants from start. These important information should be accessed easily regardless of location (C1). A structured presentation of these information in definitions is needed for the team (C9). Besides, in a running instance, participants need to synchronize their progress in time, especially in a highly-interrelated work environment, the participants not only adjust their current tasks according to other people's progress (C9), but also give feedback or guidances for one's further work (C13).

Connected entities (R2): In terms of user experience, combined with criterion C2, C4, C5, we need strongly connected entities in our supporting model. A participant has a strong willingness to switch frequently with different process definition as references (C4). It is a waste of time to go back and forth between single definition interface and the list of certain type of definition, plus it is painful to scroll the long list and locate the desired model. Simple and clear navigation between different interfaces will significantly shorten the operation time and improve user experience (C2) [Shn10].

Whereas in some circumstance that information is frequently transmitted, it should also be guaranteed that frequent switch does not increase duplicate data transmission (C7). Such an architecture need to be defined in advance of the enactment.

Communication (R3): Consider mobile network functionality with criterion C1, C3, C5, C10, we require a strong communication functionality. Mobile phone itself is basically a tool to send voice or text messages in an efficient way. Thus, a primary requirement for mobile modeling is to efficiently connect with other participants, either in an active (C3) or a passive way (C5). Since users not work in a virtual environment, the final object of mobile modeling is to achieve the real organizational goal, so every user will have the requirement to contact participants in definitions (C1). Our definitions need to offer information and operation possibilities to guide them to communicate with each other. Due to the development of mobile industry, we can utilize different type of input to enhance the communication (C10).

Cognitive load reduction (R4): We have discussed cognitive overload problem in Section 2.3.3. Regarding to criteria C1, C9 and C10, we have the requirement to reduce the cognitive load in mobile modeling. Specifically, users will not focus regularly on a definition, since in reality their attraction is easily to be caught by changing circumstance (C1). On the other hand, the method to display the architecture and interrelation of data on large screen is not applicable for mobile phone's limited screen (C9). A traditional way for displaying the information and offering interactive elements may not be possible. Also a quick method to access and input contextual information is needed in order to reduce the processing time on the mobile modeling scenario(C10).

Availability (R5): On a basis of IT service management, the availability of mobile application should be considered. Consider the communication between client and server, we need to make sure that large data transmitting process does not block the interface (C6), and reduce the synchronizing frequency as much as possible (C7). In addition, all the changes applied into local and remote database should be revertible in case of mistakes (C11). Undoubtedly, both human-failure or external failure should not stop our execution crash the application (C12).

The relationship between concluded requirements and criteria is showed in Table 3.3

In this section, firstly we introduced a detailed criteria catalog, then we utilized a scientific method to analyze the importance of each criterion with our entity definitions in informal process and the mobile modeling scenario, and finally we have a comprehensive requirement list to cover each field to support the mobile modeling scenario. In next section, we will present the concepts based on the requirement.

Table 3.3: Criteria and Requirements

Score	R1	R2	R3	R4	R5
Participant Mobility (C1)	×		×	×	
Execution Frequency (C2)		×			
Acuteness (C3)	×		×		
Correlation (C4)		×			
Acknowledgment (C5)		×	×		
Data Transmit Volume (C6)					×
Synchronization Frequency (C7)	×				×
Computing Power (C8)					
Presentation (C9)				×	
Type of Input (C10)			×	×	
Data Management (C11)					×
Fault Tolerance (C12)					×

4 Mobile Modeling Concept

In this chapter, we will prompt the concepts satisfying the requirements introduced in Section 3.3.

4.1 Extended Definitions for Informal Process Supporting Model

In this section, we will prompt some new entity definitions, i.e., *mobile entity definitions*, to extend the IPSM model which we have discussed in Section 2.3.1 on page 22. These extended entity definitions is defined with our new definition language in schema as shown in Listing 4.1. All the extended definition in this section will include mobile entity definitions.

Listing 4.1 Extended Entity Definition

```
<complexType name="tMobileEntityDefinition">
      <complexContent>
            <restriction base="ipsm:tEntityDefinition">
                  <attribute name="definitionLanguage"
                      type="http://www.uni-stuttgart.de/iaas/ipsm/mobile"/>
            </restriction>
      </complexContent>
</complexType>

<complexType name="tMobileEntityDefinitions">
      <sequence>
            <element name="MobileEntityDefinition" type="ipsm:tMobileEntityDefinition"
                minOccurs="1" maxOccurs="unbounded"></element>
            </sequence>
</complexType>
```

4.1.1 Notifications and Interactions (D1)

According previous study in Section 2.3.3 on page 25, in our mobile modeling scenario, push notification will be an efficient way to give some information to team members about what has been changed for certain entity. It also allows user to make some reaction to the modification, or in emergence directly contact the operator about details. It exploits the advantage of modern mobile devices to share real-time information. In addition, it is according to user requirements to decide which message he will receive. Subscribing interested entities, then only the notifications from these entities will be sent to user.

As well as passively receiving the notification, we utilize an operation history module to enable user to check the recent modification actively. A problem for push notifications is that there are scenarios these notifications are filtered or muted. For instance, when a user is in a meeting, he undoubtedly needs to mute his phone and focus on the discuss. In some other cases user only subscribes the topic he interested, e.g the informal process *user information analyzing*. In these cases, user will miss some updates, what user only knows is a "snapshot" when user checks the entity, and under the premise that he remembers entity detail. To solve this problem and support notification, a history interaction list is introduced, all the modifications will be created automatically, and also history state will be recorded if user clicks on it.

By a passive and an active method, participants will not miss any information related to the project, and will not be interrupted by the irrelevant information, therefore requirement **R3** is satisfied. We create the definition of notification and interactions as shown in Listing 4.2.

4.1.2 Note List (D2)

When participants collaboratively work together, it would be common if they want to leave some messages regarding to some issues in processes. The easiest case for people sitting in the same office is, just get together and have a talk if they are free. But in real business case people used to have some task in hand and have their own priorities to process these tasks. Sending email is a common alternative way for some messages, whereas it is in low efficiency. In our cases, we want both to simplified some trivial note and consider the acuteness of messages in some special case.

Thus we provide a note list to mark occurring issues and share the work progress. Firstly, the notes create a forum-like atmosphere for participants, which enables user to attach some notes to model instance, plus enable other user to respond the notes. For example, under the current running strategy instance *compare mongoDB and couchDB for setting*

Listing 4.2 Notification and Interaction Definition

```
<complexType name="tNotifiableEntityDefinitions">
      <sequence>
            <element name="MobileEntityDefinitions" type="ipsm:tMobileEntityDefinitions"
                minOccurs="0" maxOccurs="1"></element>
            <element name="Receiver" type="ipsm:tParticipants" minOccurs="0"
                maxOccurs="1" ></element>
            <element name="NotificationTitle" type="string"/>
            <element name="NotificationContent" type="string"/>
            <element name="NotificationSubContent" type="string" minOccurs="0"
                maxOccurs="1"/>
            <element name="Interaction" type="ipsm:tInteraction"></element>
      </sequence>
</complexType>

<element name="notification" type="ipsm:tNotifiableEntityDefinitions"></element>

<complexType name="tInteraction">
      <attribute name="createdAt" type="dateTime"/>
      <sequence>
            <element name="MobileEntityDefinitions" type="ipsm:tMobileEntityDefinitions"
                minOccurs="0" maxOccurs="1"></element>
            <element name="Action" type="QName"></element>
            <element name="Editor" type="ipsm:tParticipantEntityDefinition"></element>
            <element name="Target" type="QName"></element>
            <element name="SnapshotEntity"
                type="ipsm:tInteractiveEntityDefinition"></element>
      </sequence>
</complexType>

<complexType name="tInteractionList">
      <sequence>
            <element name="Interaction" type="ipsm:tInteraction" minOccurs="1"
                maxOccurs="unbounded"></element>
      </sequence>
</complexType>
```

up database, participants could give some comments on it like *consider using mongoDB 3.2 is released today, please check* or leave some technical questions that the participant do not have the idea who can resolve it, like *any suggestions on database benchmark tools?* Additionally, user could share some important information as the process going on like posting the administrator username and password to the note list.

Furthermore, the location information which is directly got from the GPS sensor, the photos instantly taken from the camera of mobile phones could be applied into note list, which will dramatically add the flexibility for participants. Imagine a user just face an

error on an automotive testing electronic device, whereas it is painful to type all the logs on a mobile screen. Sending a photo not only reduces the operation time and escalates the correctness, but also adds some contextual information like the warning light on the device, which is easy to be ignored or which is hard to describe. Additionally, as a matter of fact, we get more and more sensor on the mobile phones [1], more information could be automatically added like temperature, barometric, etc., in order to achieve a context-aware functionality as we state in Section 2.3.3 on page 25. To sum up, the purposes of note are variant, whereas they all aim for supplementing a collaborative teamwork environment and equally **R1**.

According to our analysis in Section 2.3.3 on page 25, an integration of note list with notifications (D1) could effectively improve the communication. Together with our subscribing mechanism mentioned in Section 4.1.1, the note list could fulfill the acuteness in **R3** by sending notification. This is because some information need to be spread quickly to users who concerns about subscribed entities. For instance, a note called *server configuration from 3 pm. to 4 pm.* under the strategy instance *back-end server development* could quickly notify the subscribed users about the information. The definition in IPSM schema is shown in Listing 4.3.

4.1.3 Privilege Management (D3)

Different team members need be assigned variant roles in a complicated project. Administrator required the privilege to create new goals and strategies, committers want to grant the access right to related strategy and give some feedback to executives, external specialist should be excluded from certain confidential business process to guarantee the interest of corporation. Whereas the privilege is not limited to these cases, executive in an informal process has to think about how to distribute these privileges in order to set a collaborative team, in our case, to fulfill **R1**. We will create an privilege management structure based on an role based access control which we discussed before in Section 2.3.3 on page 26. Specifically, there are four basic roles in our application.

Reader decides which user could see the definition, it is an elementary role for accessing the definition.

Writer decides that which participant could modify the definition, as well as create some notes in order to give some feedback.

[1] http://www.phonearena.com/news/Did-you-know-how-many-different-kinds-of-sensors-go-inside-a-smartphone_id57885

Listing 4.3 Note Definition

```
<complexType name="tNoteList">
      <sequence>
            <element name="Note" type="ipsm:tNoteEntityDefinition" minOccurs="1"
                maxOccurs="unbounded"></element>
      </sequence>
</complexType>

<complexType name="tNoteEntityDefinition">
      <attribute name="id" type="string"/>
      <attribute name="createdAt" type="dateTime"/>
      <attribute name="updatedAt" type="dateTime"/>
      <sequence>
            <element name="MobileEntityDefinitions" type="ipsm:tMobileEntityDefinitions"
                minOccurs="0" maxOccurs="1"></element>
            <element name="Editor" type="ipsm:tParticipantEntityDefinition"></element>
            <element name="targetNoteId" type="string" minOccurs="0"
                maxOccurs="1"></element>
            <element name="ContextualParameters" type="ipsm:tOtherParameters"
                minOccurs="0" maxOccurs="1"></element>
      </sequence>
</complexType>
```

Executor represents a role who could enact a definition to an instance, who is able to convert a strategy to an informal process. These decisions are made in a strategic level, thus an execution right is necessary in such cases.

Administrator is the most important role, which grants the right to modify the privileges for all the participants, an administrator could assign any one among these four roles to other participants.

Important to realize that we have not mentioned the right to delete a definition, in our case, delete right is granted to *executor* role. Since in a project it is important to archive completed processes and definitions for further references or modifications, deleting operations rarely happens during business process, and it should be used cautiously.

Besides the listed privileges, consider the super user in Unix system, we also need to grant a global administrator right to the executives. Such privilege is special, firstly it is a global permission, it will be applied the same content to every definition, secondly it is basically a settled value, it will be assigned in most case only once. Hence we will configure it in server side database, rather than preserving it in definition.

Applied these roles, each definition in Section 2.3.1 could assign the variant permissions to a group of participants, thus we could achieve a complete access control for the whole team in order to support **R1**. For example, the people who initiated a goal and have a

Listing 4.4 Privilege Management Definition

```
<complexType name="tPrivilegeManagement">
      <sequence>
            <element name="MobileEntityDefinitions" type="ipsm:tMobileEntityDefinitions"
                minOccurs="0" maxOccurs="1"></element>
            <element name="Reader" type="ipsm:tParticipants"/>
            <element name="Writer" type="ipsm:tParticipants"/>
            <element name="Executor" type="ipsm:tParticipants"/>
            <element name="Administrator" type="ipsm:tParticipants"/>
      </sequence>
</complexType>
```

comprehensive view of the whole project, an administrator privilege will be granted, and for the external developer in an informal process, a write role will be distributed, etc. The definition of privilege management is shown in Listing 4.4.

4.1.4 Interactive Group (D4)

As a participant in *participant list* is a significant and frequently referred entity in an informal process as show in Fig. 2.1 on page 21, there are scenarios to select multiple participants. For instance, in privilege management, project administrator needs to assign the different privileges regarding to variant definitions and people like developers, market specialists and recruiters, etc., as the scale of team increases, it would be painful to select many participants repeatedly. So a group definition is necessary in our model to enable a convenient selection functionality.

The usage of group is not limited to enhance the user experience. Inspired by note list, an information sharing capability inside a certain group is possible, which offers another dimension to share common notes which especially focus a certain group of participants. Furthermore, notice that a process could be deleted, what if the participants want to post some information permanently like database administration password in a group named *back-end server developer*. Rather than setting an auto reply in user's email box when the user is off, a simple announcement could be used like "please contact Bob about the detail of proxy server because I am on my holiday until this Friday." By this alternative way we reduce the time in writing an extra email and thus improving efficiency.

By using the definition of an interactive group as shown in Listing 4.5, we add the flexibility in permission assignment in order to support **R4** and note list attached to group improves the efficiency of work progress synchronization in terms of **R1**.

Listing 4.5 Interactive Group Definition

```
<complexType name="tInteractiveGroup">
      <sequence>
            <element name="MobileEntityDefinitions" type="ipsm:tMobileEntityDefinitions"
                minOccurs="0" maxOccurs="1"></element>
            <element name="Participants" type="ipsm:tParticipants"></element>
            <element name="NoteList" type="ipsm:tNoteList"></element>
            <element name="PrivilegeManagement" type="ipsm:tPrivilegeManagement"/>
      </sequence>
</complexType>
```

4.1.5 Interactive Participant (D5)

A fundamental function of mobile phones is phone call function, which is an efficiency way to communicate with other people, it is also a reason why most of people bring mobile phones everyday. In an informal process, it is common for participants to have phone call to respond an unpredictable emergencies, have a quick feedback or discuss processing detail.

Though we have created the note list (D2) for sharing the information, considering a place with a bad Internet connection, users still have to use text messages on some special occasions. Moreover, the note list (D2) is a supplement to text message, it does not aim for peer to peer messaging function. In contrast, modern phones have integrated this native function much better. Another fact is that when discuss trivial topic like a reminder, it is better to use a private channel rather than interrupting other people.

Last but not least, email function is also needed, the reasons are, firstly, the email is counted more official, according to the law in different country or area, it could be a legal evidence or a standard commercial document. Secondly, user in a mobile context has a requirement to state a complex or complicated problem with relatively long structured article.

We extend the participant definition in informal process by including all the contact information of participants, which is shown in Listing 4.6 . Therefore, we create multiple communication channels for supporting model in addition to share the information through updating entities in IPSM. These official and private communication channels become supplements of **R3**.

Listing 4.6 Participant Definition

```
<complexType name="tParticipantEntityDefinition">
      <sequence>
            <element name="MobileEntityDefinitions" type="ipsm:tMobileEntityDefinitions"
                minOccurs="0" maxOccurs="1"></element>
            <element name="telephoneNumber" type="ipsm:tPhoneNumber"></element>
            <element name="emailAddress" type="ipsm:tEmailAddress"></element>
            <element name="website" type="string"></element>
      </sequence>
</complexType>
```

4.2 Mobile Modeling Supporting Features

Feature is a prominent attribute or aspect, they are some specification which will be implemented in mobile modeling applications. In order to fulfill the requirements **R2**, **R5**, we have the following mobile modeling support features in addition to the extended concepts.

4.2.1 Linking Definitions (F1)

We have interrelated entities and each entities will have a lot of attributes as shown in Section 2.3.1 on page 22. In our mobile modeling scenario, all the names or references displayed on the screen are clickable, and which will ceaselessly redirect to the their detail interface. Regarding to a view of one intention, all the related contexts, and all target strategies is selectable, user clicks on the name, e.g., context *core team member* in the intention *team building*, it will lead to the corresponding page. Equally important, a backward navigation need to be supported as well. Theoretically, such navigation stack could store as many pages as possible, that means, users could click and jump without any limitation if they see what they curious about. Furthermore, we will deploy a drawer to categorize different kinds of our model concepts. It acts as the a navigation bar in modern web applications. By this drawer user can redirect to the main list of other models.

Integrating these ways in the application will give user an escalate experience to navigate wherever he wants, which support **R4**. In addition to that, it creates an intuitive concept, if more information is needed, user just need to click on it. By above features, **R2** is satisfied. To implement such linking models, a well designed router is required for the app, we will comprehensively discuss in Section 5.2.2.

4.2.2 Asynchronous Caching for Modeling State (F2)

Previous **R5** has referred to both human failure and network failure, thus a method to tolerant these failures need to be developed. An asynchronous caching mechanism which caches the modification in the client side during modeling will be used to address both aspects.

To be more precise, as user enters into one entity like intention or strategy, if he got the permission, some modeling modification could be made, at this moment, all the operations happened only in client side, any changes are not made to related entities or remote servers, only after he clicks the save button, the changes will asynchronously applied to both the server side by communication and related definitions on client side.

Human and network failure are both addressed in this case. Firstly, user could undo any changes he had made just by clicking a **revert** button, thus no effect takes place. Secondly, if any Internet error happened, the model will remain the edited state as long as user stayed on such screen, any further operation will not be suspended. only after the user receives a callback message indicating that the server successfully changed the value, new modification takes place.

It is needed to consider a competitive scenario where two users send modeling request at almost same time. We records the latest one received on server, but the both modified states are recorded in the interaction list as we mentioned in Listing 4.2 on page 39.

In our modeling process any modification will strongly affect the related models and send a notification to participants. Therefore we have not adopted a popular way in modern software application forwarding any changes directly to server. Our solution is firstly caching the edited data in database, then records all the modification in operation history. Therefore, by this feature we assure availability **R5** of our mobile modeling scenario.

In this chapter, all the concepts focuses on one or two requirements, to sum up, we made the following table Table 4.1 to make a better illustration.

Table 4.1: Features and Requirements

	R1	R2	R3	R4	R5
Notifications and Interactions (D1)			×		
Note List (D2)	×		×		
Privilege Management (D3)	×				
Interactive Group (D4)	×			×	
Interactive Participant (D5)			×		
Linking Definitions (F1)		×		×	
Asynchronous Cachingfor Modeling State (F2)					×

5 Mobile Modeling Case Study

In this chapter, firstly, we will introduce the fundamental framework. Hereafter, we detail the implementation for concluded concepts from Chapter 4. Then we elaborate the application structure from perspective of software engineering. After that, we present the use case about our motivation scenario to illustrate the different models in detail and make a extensive view of the modeling concept. Finally we conclude the added value in supporting mobile modeling scenario from analysis of the use case.

5.1 Application Framework

As framework decides the basic architecture and features of the application, affects the efficiency during run-time execution, we will discuss the design art on how to implement our concepts in this section.

5.1.1 Basic Platform

Mobile operation system range from Android, iOS, Windows Phone OS, to some newest innovation like Sailfish OS and Ubuntu OS. Among them, Android and iOS almost entirely dominant smart phone market in the world. Whereas Android and iOS have totally different development environments which require multiple skills like API knowledge and programming languages regarding to them. Recent years, hybrid mobile app development has attracted developer's attention since it uses web technology to develop the mobile application which is compatible in both Android and iOS platform. It shortens the learning curve for two platforms and enables creating a consistent user interface and reusable functions.

Native code is compiled code, it is faster than interpreted language [CL11]. Compared with JavaScript powered apps running in a web browser, native app gets closer to phones processor, sensor, memory, etc. By revoking platform native functions, it not only improves the efficiency of compiling and processing speed, but also integrates the capabilities on hardware which can not be achieved by web-based hybrid app.

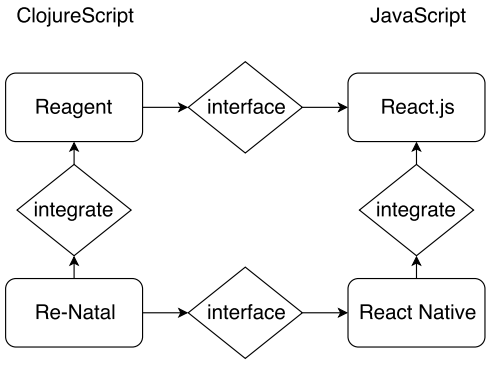


Figure 5.1: Re-Natal and React Native

5.1.2 React Native

Particularly, React Native compiles JavaScript code to native code. It makes no performance compromise compared with app built with native tools like xCode, and provides the access to most of mobile hardware. Additionally it combines both the advantages of hybrid app and native app. React Native also has an open source code repository, a large developer community, and most importantly, it enables an component developing style for mobile applications. Regarding to our model, there are variant references, elements, attributes for each entity, and they are connected to each other. On this point, React Native would be the best choice for us compared with hybrid app building tools and native app building tools.

As we stated in Section 2.3.2, our mobile modeling application rely on Winery, OpenTOSCA, etc., as contextual interfaces, it acts in the same level with a web modeling application. Therefore, we will consistently use ClojureScript[1] as our programming language which is already used before. An advantage for using ClojureScript is, it uses atomic value in storage, thus we could directly store the state value into a global state machine. In contrast, under JavaScript environment developer have to use additional state container like Redux[2] or Flux[3] to maintain states.

In order to use ClojureScript with React Native, we will adopt Reagent as an interface to React.js, and use Re-Natal[4] to integrate Reagent[5] as an interface to React Native. Their relationship could be seen in the Fig. 5.1

[1]https://clojurescript.org/
[2]http://redux.js.org/
[3]https://facebook.github.io/flux/
[4]https://github.com/drapanjanas/re-natal
[5]https://reagent-project.github.io/

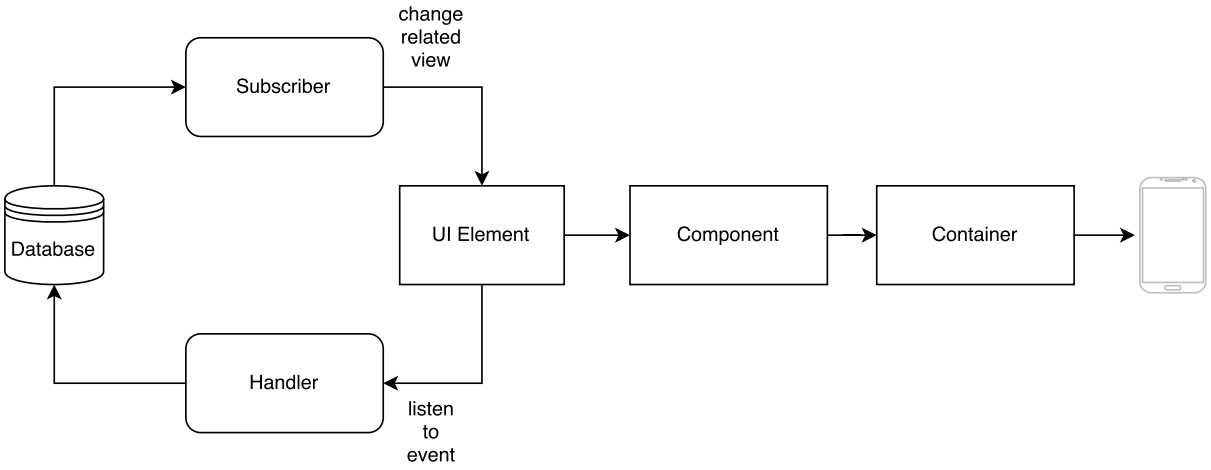


Figure 5.2: Application Architecture

5.1.3 Application Architecture

We have deploy our project file in a Gitlab repository[6]. It basic architecture could be seen on Fig. 5.2. It applys re-frame[7] two-way data binding and React components together.

We use a two-way binding to remove the latency between user interfaces and data models. Specifically, as user successfully updates a view, registered handlers will apply the changes to database, as an consequence, any subscriber listening to the certain value will capture this event and broadcasts the modifications to related views. Components is the fundamental of React Native, they are reusable UI elements. According to the complexity, we have three level of components. The subscribed data flow is directed from UI components to customized components and finally to containers, and then user see the change. In turn a interaction event is handled by a similar way.

5.2 Implementation of Concepts

5.2.1 Extended Definition Implementation

In this part, we will give the technical details about the implementation of the extended definition we have discussed in Section 4.1 on page 37. All the implementations are

[6]https://gitlab.com/hanwencheng/co-act-mobile
[7]https://github.com/Day8/re-frame

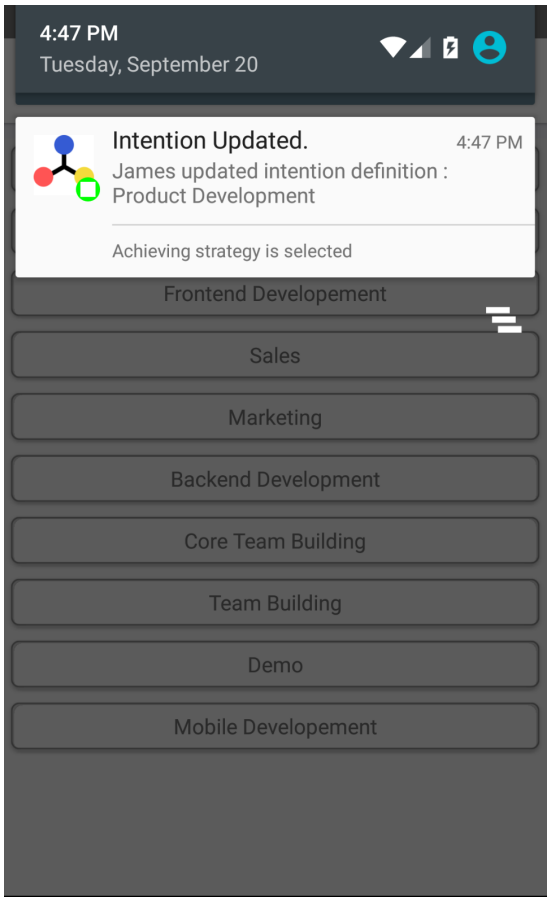


Figure 5.3: Notification Implementation

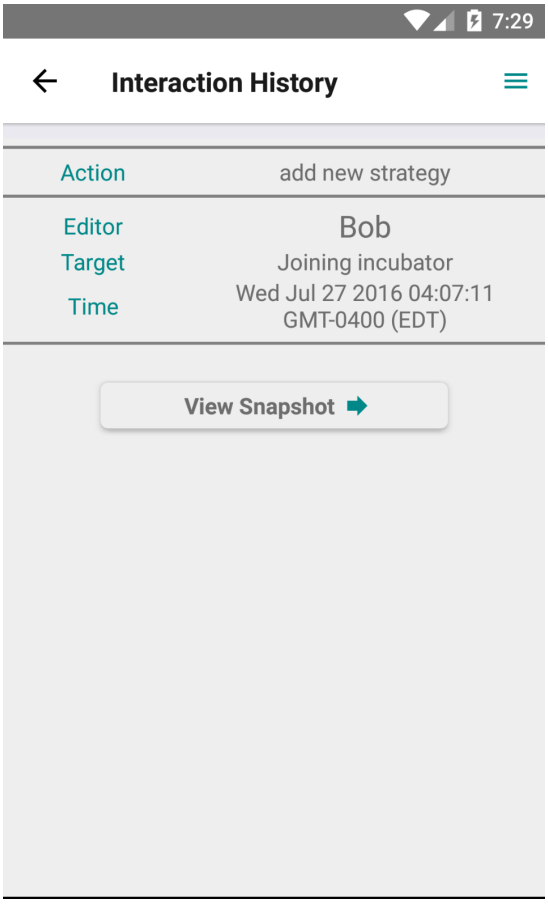


Figure 5.4: Interaction Implementation

taken place in an real Android application, and we will use application screenshots as supplement in terms of better illustration.

Notifications and Interactions (D1*)

We utilize the notification function in Android. When a subscribed entity has been modified, a push notification will direct be prompted immediately on the top of screen with its title and sub-title as in Fig. 5.3. In addition, when user drag down the notification list, the main content will be shown. These modifications are categorized as interactions, which is records in a separate record list, it store essential information and a “snapshot” of the modified state as shown in Fig. 5.4

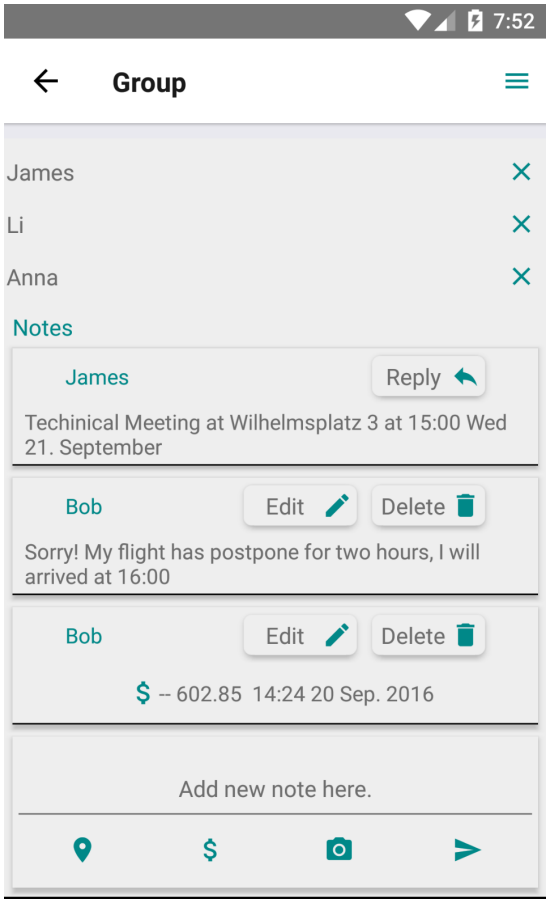


Figure 5.5: Note List Implementation

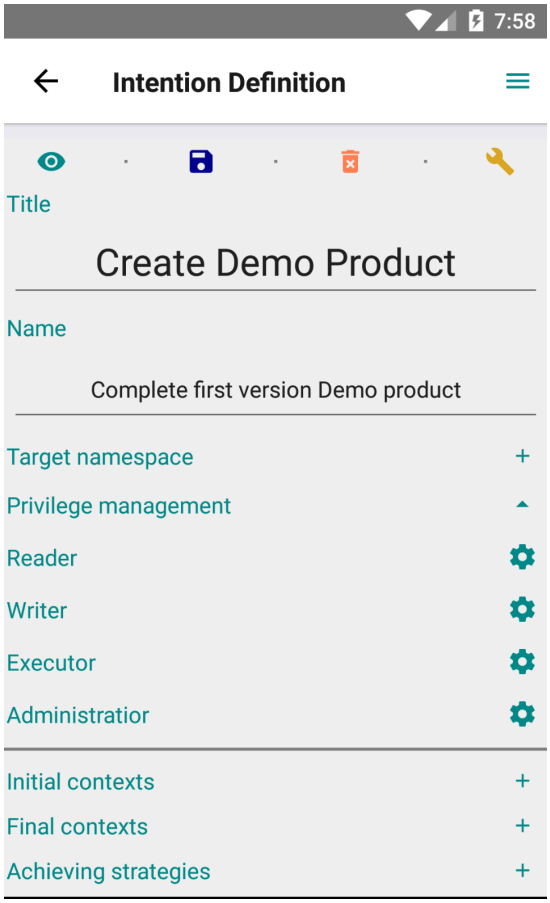


Figure 5.6: Privilege Management

Note List (D2*)

Note lists are composed of notes which are created by participants, they are attached to all the instance descriptors and interactive groups. User having permission could post new note, edit his previous note, and reply to other user's notes. We offer a button to get a quick access for GPS location, and convert these raw information from latitude and longitude to text by Google Geocoding [8]. Additionally, by some customized button we offer variant way to attach contextual information to a note. For example, here we use a Bitcoin button, by clicking user could directly get the current price of Bitcoin[9] and create a post with the value as shown in Fig. 5.5. vre

[8]https://developers.google.com/maps/documentation/geocoding/intro
[9]https://bitcoin.org/

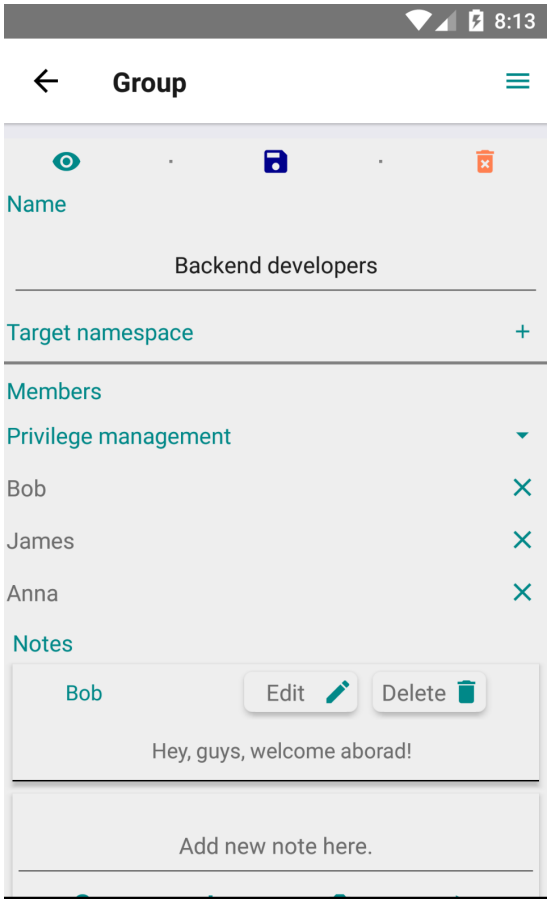


Figure 5.7: Group Implementation

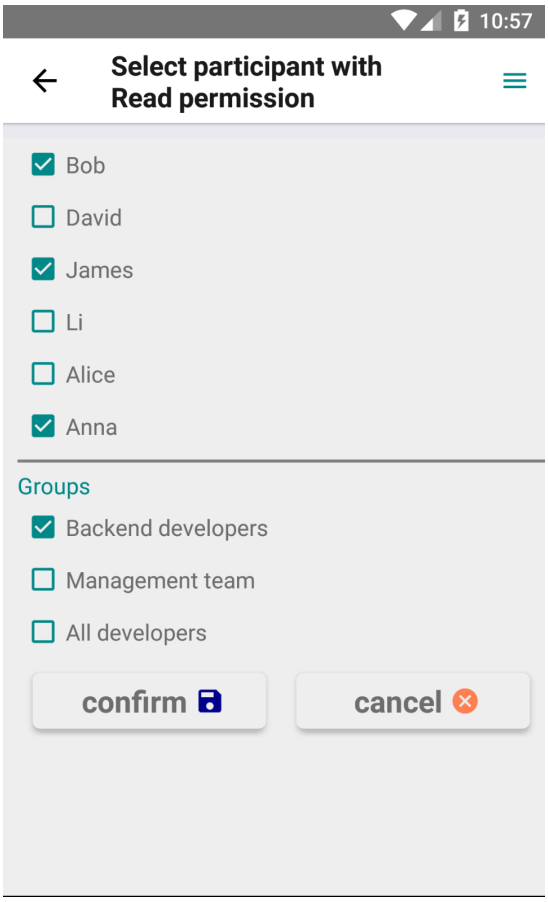


Figure 5.8: Selecting Permission

Privilege Management (D3*)

We embedded a privilege management control panel to each interactive entity definition interface. It is only visible for the user who has an administrator role. Since user's screen size is limited, the space should be reserved for core properties in terms of representation, we will build the detail control into a drop-down menu as shown in Fig. 5.6

Interactive Group (D4*)

We create an definition interface for interactive groups as shown in Fig. 5.7, all the members could be clearly seen from group interface. User could also post note inside the group for sharing information as we mentioned before. It also affect the privilege selection as shown in Fig. A.10. If user select a group, all the group member in the list will be chosen automatically.

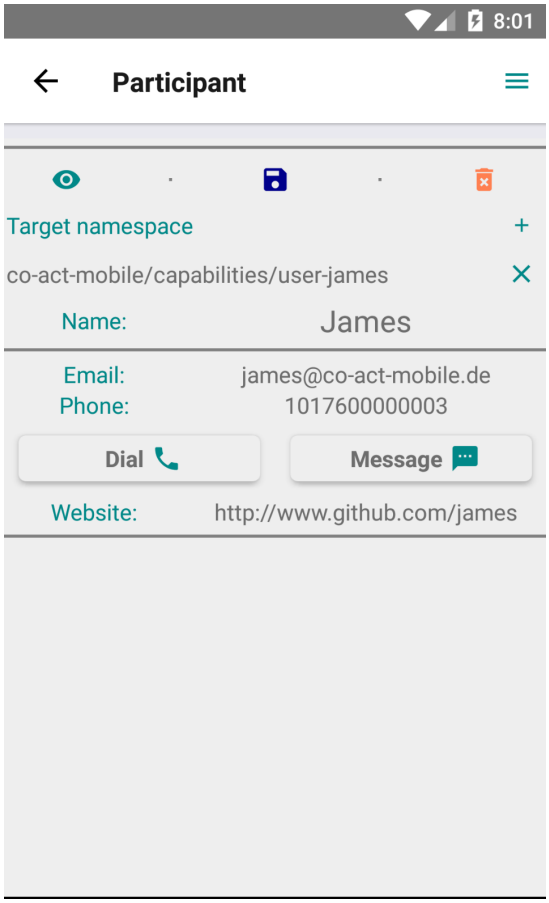


Figure 5.9: Participant Implementation

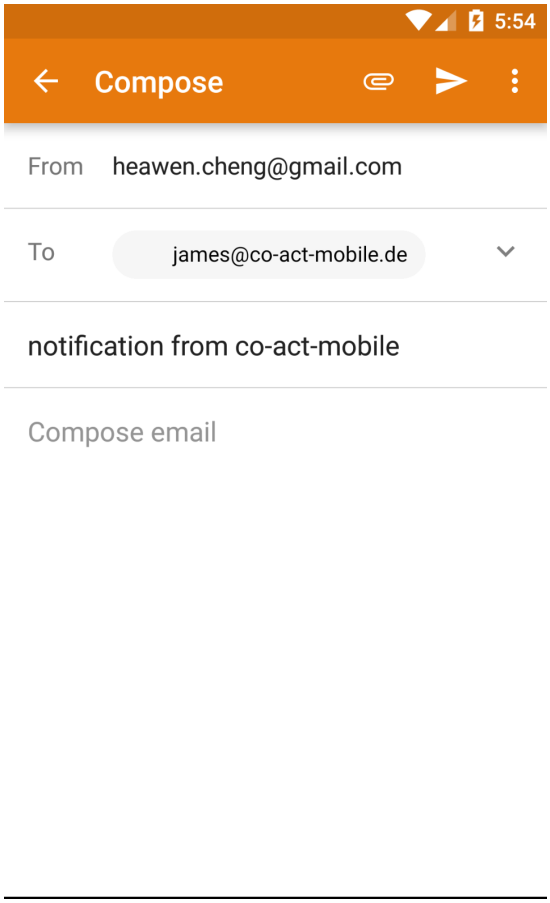


Figure 5.10: Pop-up Email Function

Interactive Participant (D5*)

The participant profile page is shown in Fig. 5.9. Once user clicks on certain information, a correspond action will take place. Specifically, if user clicks on the email button, email application will be directly initialed with a target email address as shown in Fig. 5.10, if user clicks on the text button, then a text writing screen will be occurred with a filled target number. Furthermore, the related Internet address like a social network page or a personal website will also be shown here as a reference. A simple click will directly lead to phone's embedded browser jumping to the page.

5.2.2 Mobil Modeling Supporting Features Implementation

In order to realize the supporting features we mentioned in Section 4.2 on page 44, we will illustrate our router and navigation mechanism in this part.

A router stack has been created to cache the content displayed in the screen. Whenever user touches the link and be directed to another page, stack changes. The key information of the target entity page, and some contextual data like *group* and *title* together compose a navigation state object, and this object will be pushed into a First In First Out (FIFO) stack as shown in Fig. 5.11.

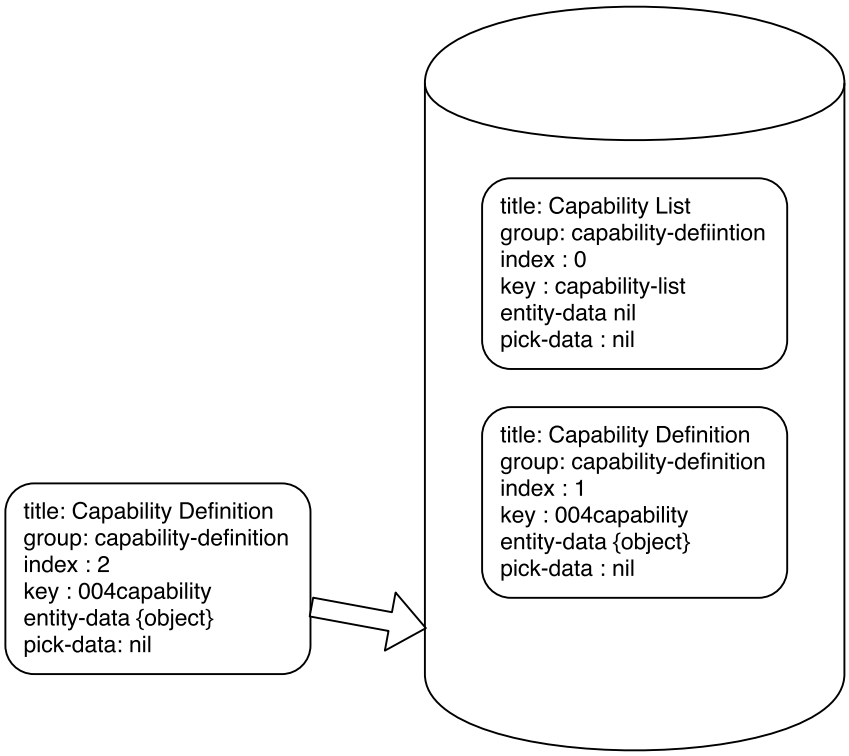


Figure 5.11: Router Stack

If user wants to open another interface and clear the stack, our application revokes a redirect function, which just empties the stack and pushes the new state into stack. The redirect function is the underlying element enabling our navigator, which should include the entries of all the entity categories. In mobile application design, small navigator could be implement horizontally, which is a navigation bar on the top or bottom of the screen. Whereas in our scenario, we have variant kinds of definitions including identifiable entity definitions and extended definitions, so we will propose a vertical navigation bar on the right side of the screen. Furthermore, the screen of mobile phones are normally limited due to its portable nature, thus we have designed a drawer navigator which is hidden during normal usage, and could be called again by using a swipe gesture. As an consequence, we need to design a animation on the navigator. Align with the width of our main screen and the drawer component, we created a function to detect the swipe gesture. We define the width of drawer as 250 pixels, assign the animation duration as 200 milliseconds, and our gesture detector will

capture any horizontal movement gesture which is horizontally larger than 20 pixel. Furthermore, drawer is a top level component, which could be called in any page by predefined gesture, as shown in Fig. 5.12, and Fig. 5.13. By the design of drawer and router, we implement our linking model feature (F1).

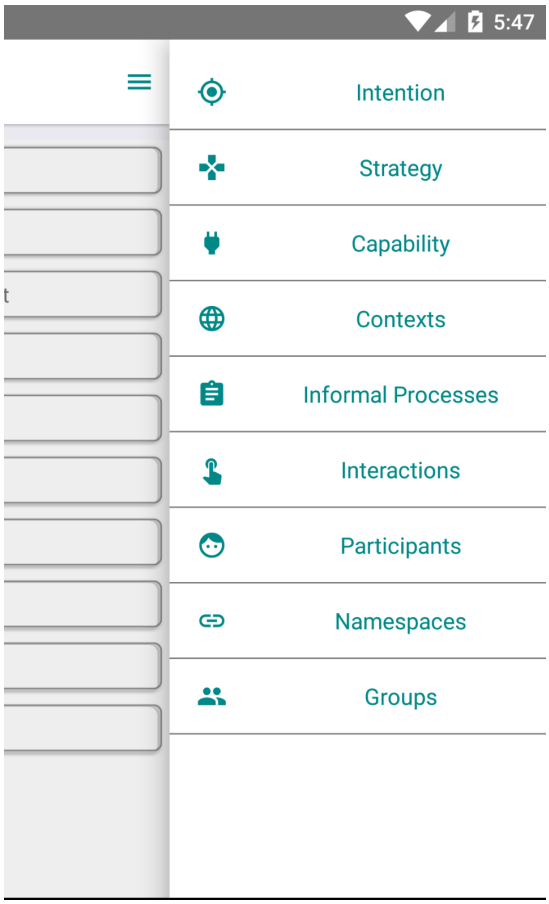


Figure 5.12: Drawer on Intention Definition List

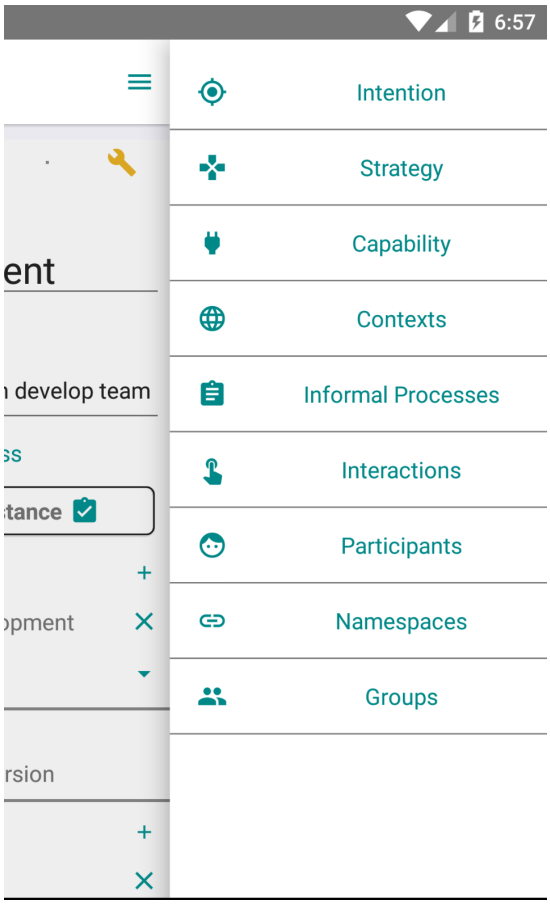


Figure 5.13: Drawer on Strategy Definition

Router controls the direction logic in the application, allow user to perform tasks in the consistent sequence across similar conditions [Shn10], and offer a navigator to enable shortcuts in operation. The stack not only stores the content of current page, it could also act as a cache to fulfil our asynchronous caching feature (F2). Since any modification related to modified entities are stored in the stack, only after receiving the success callback from server we apply the modifications to local database, then update the stack. If the HTTP request failed, any changes only remain in stack. User could just stay at current or new page waiting for the next request, or user could revert the modification since any change has not been settled yet. Thanks to the stack, we may embed save and revert function into interactive entity definitions.

5.3 Concept Validation on Motivation Scenario

To better illustrate our concepts and implementations, we will validate them based on the motivation scenario as we discussed in Section 2.2.3 on page 19. The supplement figure from Android application are placed in Appendix A.

BlockTech is a web technology company based in Berlin, with developers team settled in Stuttgart, aiming for creating a Internet financial platform, they have just set up the team and completed a business plan. Firstly chief executive officer (referred as *executive* below) creates a context definition as an initial context to acknowledge current status with whole team as shown in Fig. A.1.

After discussion with the team, executive has created three goals for the current situation as listed in Fig. A.3, which are raising the money from market or joining an incubator, making a demo product as start, and trying to persuade other people to join the team. Now executive starts to create some intentions in the application with detail as shown in Fig. A.4.

After thinking about the current status of team, executive has created some capabilities which refer to the resources existing in the current team. One capability of mobile developers including Android and iOS developer capabilities is shown in Fig. A.2

Product is a core element for technology company, so team leader now is thinking about the strategies to complete the product based on the current owned capabilities. As a consequence, executive creates different strategies aiming for product development and specifies more intentions as contained sub-intentions. One strategy of *developing product with company own developer team* is shown in Fig. A.5 and Fig. A.6

As all the models are almost created, executive now starts to open privilege management panel as shown in Fig. A.7 and assigns the roles for different team members as shown in Fig. A.8. For example, the administrator privilege will be assigned to the most experienced technical expert, who lived in Stuttgart.

Since it is tedious to assign view roles to all the team member now, executive creates several interactive group to reduce the operations in privilege management panel. Here, Fig. A.9 is created for back-end developer, and Fig. A.10 illustrate that whenever a group is selected, all the member in that group will automatically be chosen by application.

Team members now receive the notifications that they have been assign to certain group. Then technical expert decides the achieving strategies for product development intention. Controlling the code with company's own developer team will produce an more flexibility and quality management, thus technical expert select this strategy from the list as shown in Fig. A.11. To be mentioned here, more than one strategy could be selected.

Then all related intentions attached to the strategy are automatically added to current intention as shown in Fig. A.12, target intentions has been assigned to achieving strategies as shown in Fig. A.13. Technical expert could direct see the result as we have enabled two way binding data models.

As shown in Fig. A.14 and Fig. A.15, executive instantly receives the modification by a push notification on his phone when he is just about to check in his flight at Berlin airport, developers who are just preparing the meeting in Stuttgart also receive the notification, as they also have view privileges and have subscribed this intention.

Executive now sits at the waiting room in the airport, it seems that his flight will be later for two hours, so he posts a note in the group as shown in Fig. A.16, lets the team in Stuttgart postpone the meeting time. Also executive reminds that an external technical expert need to be invited to the meeting, so he dials the leader technical expert directly by the phone function which is integrated into our application as shown in Fig. A.17. After communication, executive now decides to create a new team member, and assign a reader role to external expert for accessing the *product development* intention definition and relate strategy definitions.

On the flight, executive reviews all the processes, and creates new final context for first phrase intention. In addition, by checking the team members email address stored in the application, he can start writing and caching a new email by clicking on the email link on participant detail page, title and address will be automatically applied to mobile phone internal email application.

After get off the flight, executive synchronizes his modification to server, then all the changes he has made on the flight can be seen by other team members. Later executive has arrived at meeting point in Stuttgart, whereas he does not seen other team members here. Executive then posts his location in the note list by clicking the embedded location button to geographically encode his latitude and longitude from service provider, as shown in Fig. A.18. Since there are two *Wilhelm Platz* in Stuttgart, by geocoding the detail other team members eventually realize that there was some misunderstood.

On the way to a meeting room, executive posts the current price of Bitcoin as a reference for meeting as shown in Fig. A.19. After discussion the requirement with team members, executive decides the final version intention definition *product development* as shown in Fig. A.20 and Fig. A.21 As technical expert converts related *software developing* strategy to informal process definition by clicking the convert button in the strategy as shown in Fig. A.22. The generated informal process definition is shown in Fig. A.23

Finally, executive decides to initialize the intention by simply clicking the tool button in the top right as shown in A.26. Consequently, our application automatically initializes related entities like strategy definitions, capacity definitions, and converted informal process definitions as shown in Fig. A.24 and Fig. A.25. In addition, technical expert also

sets the state of informal process instance from *preparing* to *running*. As a consequence, its original definition now will have the implementation instance reference.

We have offered instantiation function to transform a definition to an instance, we also supply extracting a new definition from a modified instance. This extract button is provided for each instance on the top right corner of the screen as shown in Fig. A.27, which replaces the tool icon in definition entity.

5.4 Discussion

As we state in Section 2.3.3 on page 27, added value is the improvement for a process activity when supported by mobile apps. In this section, we will analysis the aforementioned use case, and propose the added value in our work compared with traditional desktop and web application from different aspect.

5.4.1 Time-involving Definitions (V1)

As a viewer in informal processes, users only knows the updates when he opens the application on desktop, he will only be ware of the modifications under condition that he remembers the previous state of models. Also, he will have no clue on when the changes took place or who edited the model. In reality, some participants may still working with out-dated process though the model has already been updated. For instance, in our previous use case, team members in Stuttgart would not aware that all the models had been created by executive, and had no idea what changes had been made before.

Due to the mobile nature of application, one could receive push notification (D1) about all the updates on subscribed model, participants will have all updating information in real time. Equally, viewer can check the operation history (D1) list to see what happened in the past if he is unable to receive notification previously. Furthermore, by saving the operation and the previous states in history, we could distinctly know the information of an entity at certain point in time.

As a matter of fact, we have added a time dimension to all of the models. Time becomes a crucial part of the supporting model, as it separates the states of entity according to operations and prevent conflict. Time stamp also exist in note List (D2) according to Listing 4.3 on page 41. In viewer's mind, it will be unambiguous that when and how the modification happened, and when the business is addressed by executive. These extra information will also act as a supplement for further execution on the model.

5.4.2 Location-independent Modeling (V2)

Regarding to executives in informal process, traditional applications are only available on desktop in office. If there are any ideas to update the model, executive has to mark down them on notebooks, and edit it after he arrives at office or home. In such case not only omissions and mistakes may occur, but it may also postpone the operations to be handled. In our case, executive was still configuring the model during his trip to Stuttgart. Without the mobile modeling support, executive would not be able to add a new team member when he was waiting for his flight. As an consequence, it was pretty hard for the external technical expert to have a general view about the project before meeting.

With mobile app integration, executive is able to modify the models regardless of where he is. When user is not in workplaces, any modification is fully-supported and it is easy to operate and locate quickly with an simple and intuitive design (F1). Tiny modification like changing the priority level may just take a few seconds. In particular situation without Internet connection, it is still possible to caching changes in our definitions, since later executive could synchronize these changes again by a simple click (F2). By this we achieve a location-independent modeling, which means the modeling task could be accessed and updated regardless of location, rather than only under certain conditions.

5.4.3 Interrelated Participants (V3)

Distributed team members have a loose interrelationship, since there are always gaps between their knowledge and information regarding to the progress of work. The information acknowledgment is strongly depends on traditional communication method, like telephone and face-to-face meeting. Our interactive participant (D4), interactive group (D5) and privilege management (D3) provide a hybrid solution to this problem.

Interactive participant definition (D3) enables different communication channels by integration with phone basic functionalities, including phone call, text message, and email. Thus participants can easily be accessed. In addition, these participant profiles will form a business contact database, which will be separated with daily life contacts in mobile phones. In our scenario, executive shared the flight delay message (D2) to the group (D4), and by posting his real-time location from mobile phone's embedded GPS, other team member recognized where he is. In addition, developers could post the announcements and notices like meeting essentials afterwards. On the other hand, these interrelationships are controllable, since we have a flexible privilege management, which could assign the privilege to different people (D3). In previous scenario, executive

Table 5.1: Concepts and Added Values

	D1	D2	D3	D4	D5	F1	F2
Time-involving Definitions (V1)	×	×					
Location-independent Modeling (V2)						×	×
Interrelated Participants (V3)		×	×	×	×		
Context Aware Definitions (V4)		×					

only assigned the a viewer privilege to the external expert. To sum up, our concepts will enable participants to be involved into business process more actively.

5.4.4 Context Aware Definitions (V4)

In original entity definitions user can not append additional information. Only the preserved element according to IPSM schema will be maintained, plus contextual information is limited to textual presentation. The only way to add extra contextual information is to add more sentences to an original property. In our previous scenario, *Wilhelm Platz* was not accurate and caused misunderstanding.

Our concept note list (D2) offers such flexibility regarding to this problem. Participants now could pin extra contextual information to entity instances like intention instances, strategy instances, etc. They can also broadcast instant messages to a group of people. The information to be shared is not limited to a text, but could also be a location, an image, etc. In aforementioned case, executive directly added the real-time price of Bitcoin to the note list as a reference. Similarly the note list could be customized due to the focus of organization. By this method, we achieve a context-aware definition for both modeling and further execution.

We summarize relationship of added values and concepts in Table 5.1

6 Conclusion and Future Work

Conclusion

Informal process modeling is difficult compared with traditional business activities, since it relates to flexible structures and unpredictable behaviors. In this work we firstly study the background and propose a mobile modeling scenario in Chapter 2 on page 17, and use scientific method to evaluate the criteria catalog based on the scenario and definitions of informal process entities, thus getting requirements in Chapter 3 on page 29.

In Chapter 4 on page 37, we have the *extended definitions for informal process supporting model* to satisfy requirements. The extended concept includes notifications and interactions (D1), note list (D2), privilege management (D3), interactive group (D4) and interactive participant (D5). In addition to that, *mobile modeling supporting features* are introduced as supplements to fulfill the requirements, which are linking definitions(F1) and asynchronous caching for modeling state (F2).

In Chapter 5 on page 47, we present a case study on the aforementioned concepts. Firstly, we elaborate the framework of implementing mobile application, and then present the detail implementation of the extended definitions and features from Chapter 4. After that we validate a real use case on our motivation scenario, and then discuss about how the mobile supporting model could benefit our business process, i.e., time-involving definitions (V1), location-independent modeling (V2), interrelated participants (V3) and context aware definitions (V4). All of these added value are contributed by our concepts as an enhancement compared with traditional use scenarios.

We implement the modeling concept in mobile application from scratch to end. Firstly we studied informal process supporting model and schema, then designed the application work-flow and draw interface sketch with professional application design tools[1], and finally work with code, made an real Android application, plus we also prepared on iOS

[1] https://www.sketchapp.com/

platform with same view structure and data models which enabling future developing. We have deploy all the source code with Android and iOS on a Gitlab Repository[2].

Future work

Mobile application is one element in informal process modeling, application support on other platform like desktop and back-end server need to be set up and developed, in addition, modeling is just one step in terms of supporting the whole informal process as shown in Fig. 2.3 on page 24, further work like *informal process execution* and *informal process recommendations generation* will need to be integrated.

In terms of mobile application software engineering, it is a emerging field which contribute a lot to both business process and information technology. It has great possibilities in improving nowadays business process. Though we have proposed a way to enable modeling informal processes, there are still many potential capabilities to be explored as mobile technology is continuing advancing. We believe that eventually the concept of informal process will benefit from it.

[2]https://gitlab.com/hanwencheng/co-act-mobile

A Appendix

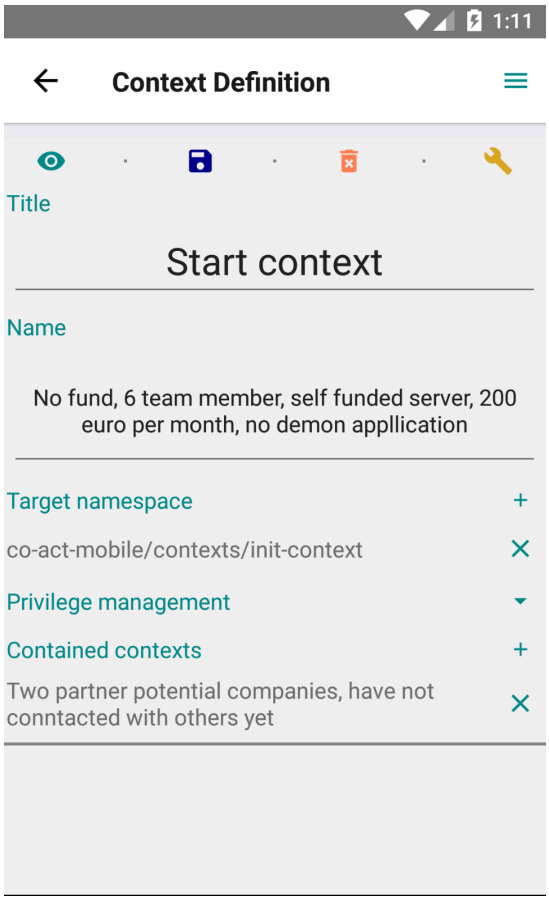


Figure A.1: Initial Context Definition

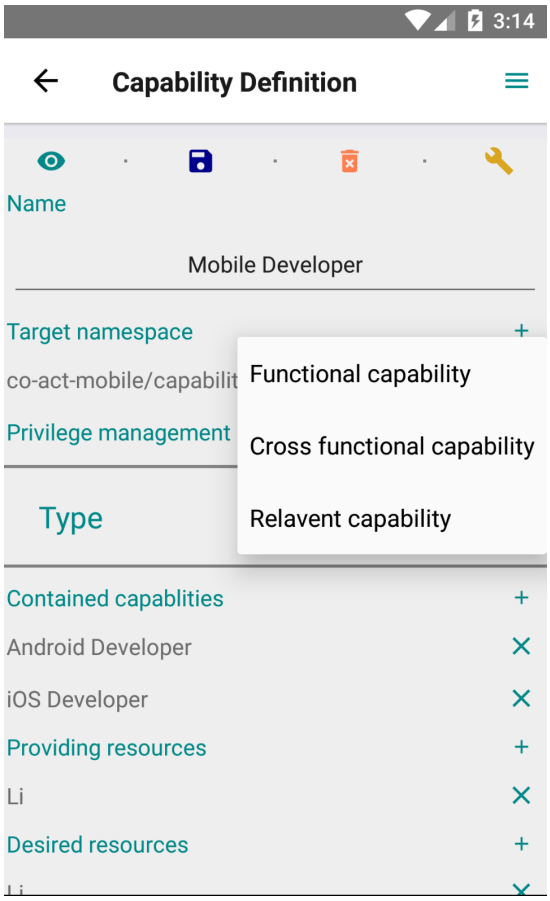


Figure A.2: Capability Definition

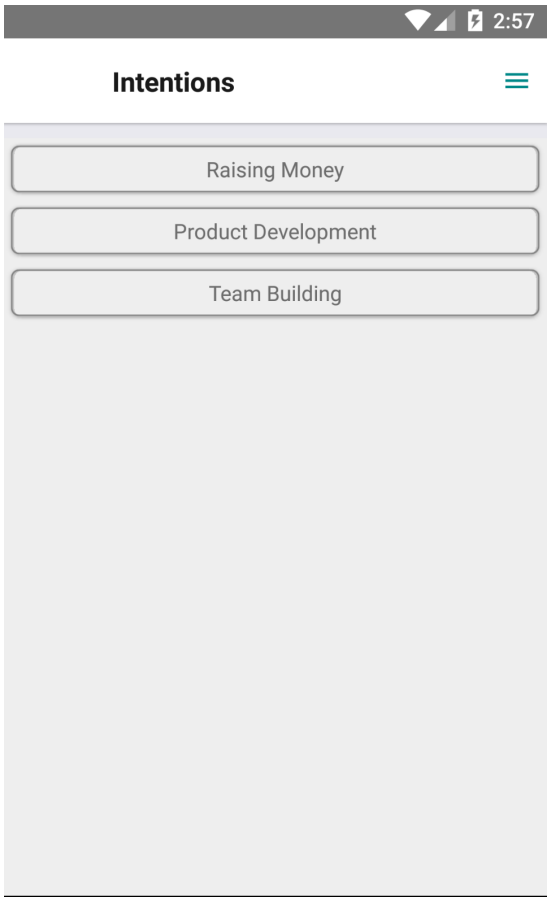


Figure A.3: Intention Definition List

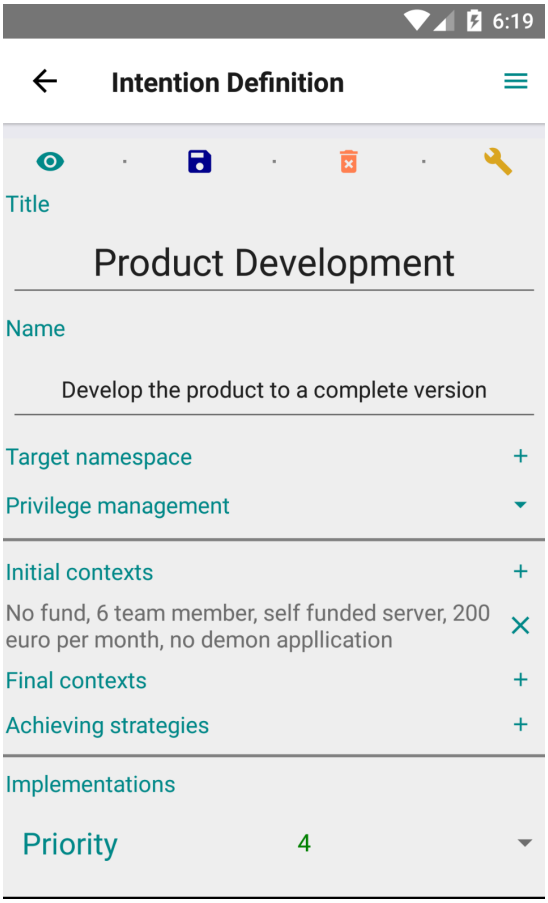


Figure A.4: Initial Intention Definition

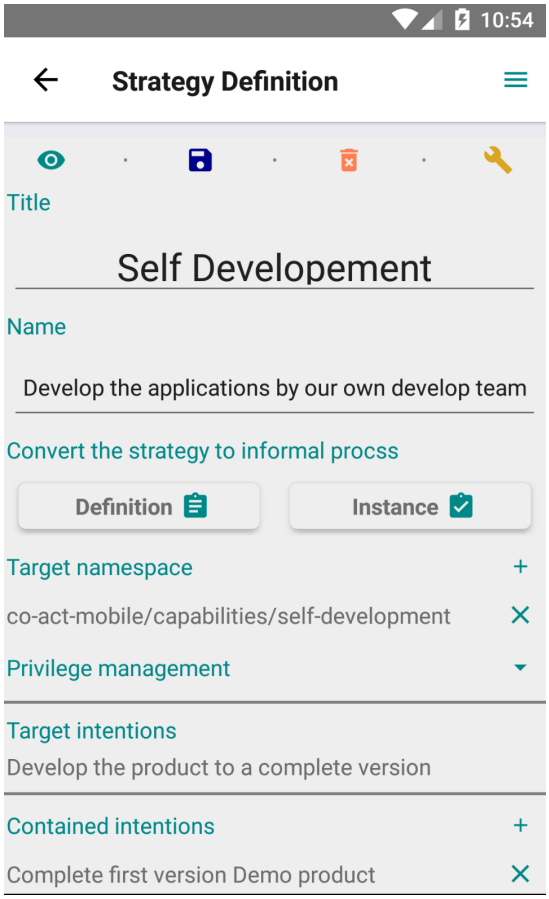


Figure A.5: Strategy Definition 1

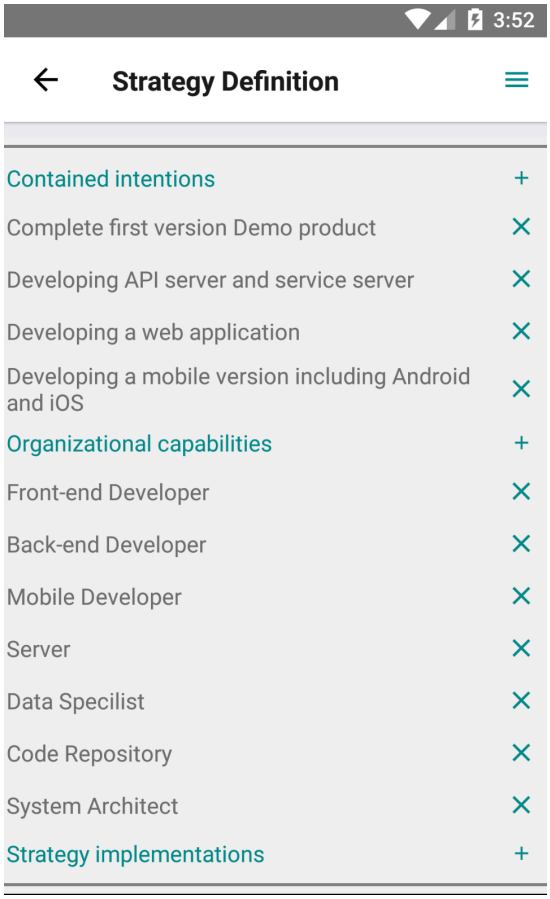


Figure A.6: Strategy Definition 2

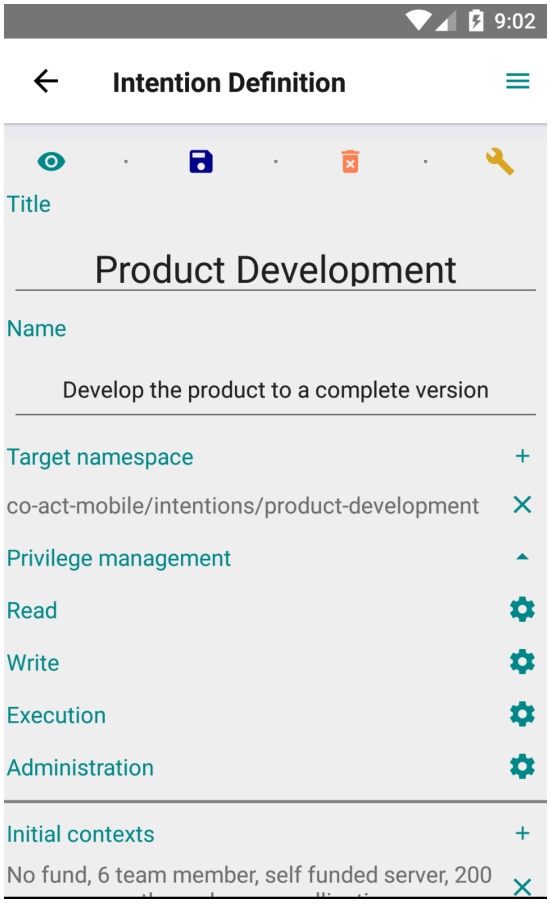


Figure A.7: Expanded Privilege Panel

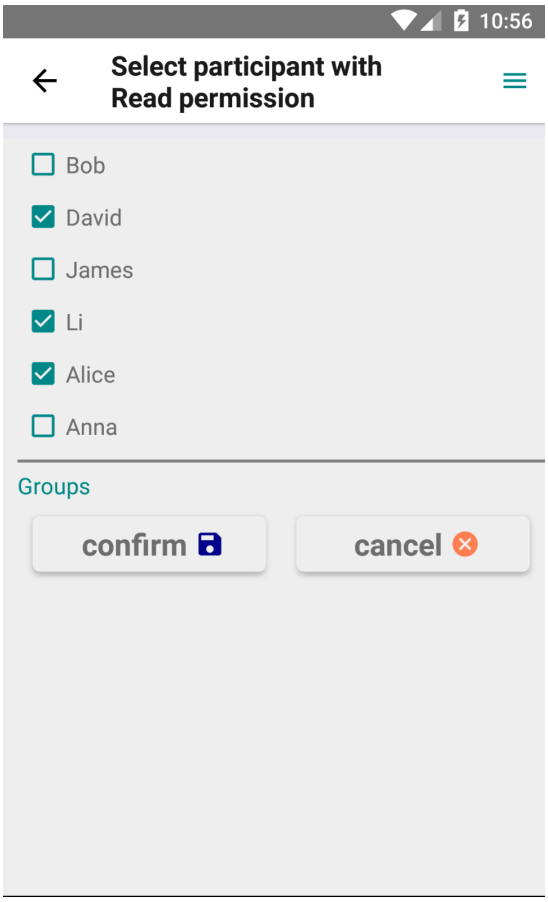


Figure A.8: Assigning Privileges

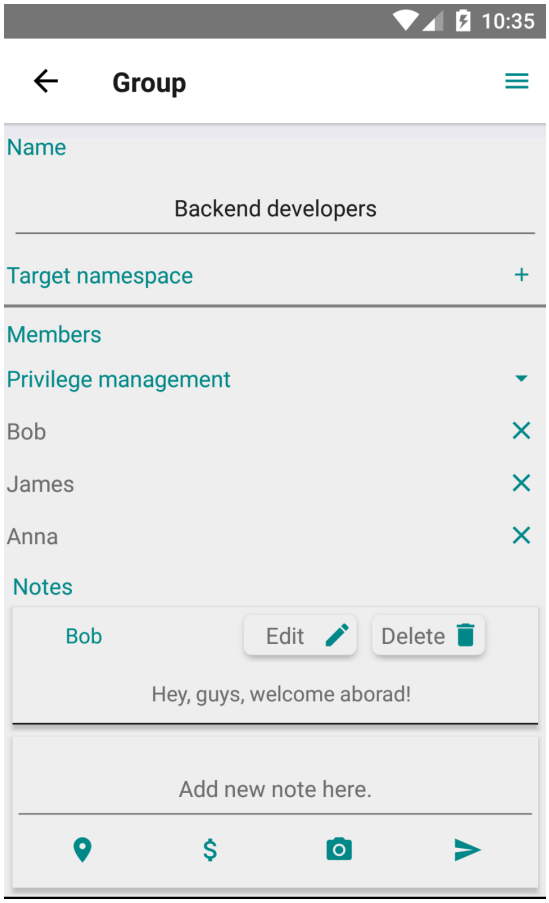


Figure A.9: Creating Group

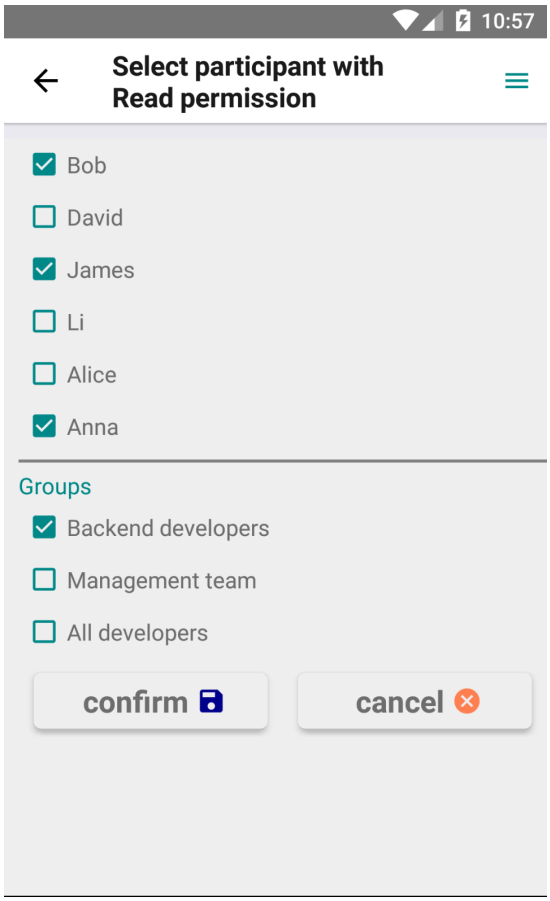


Figure A.10: Selecting Group

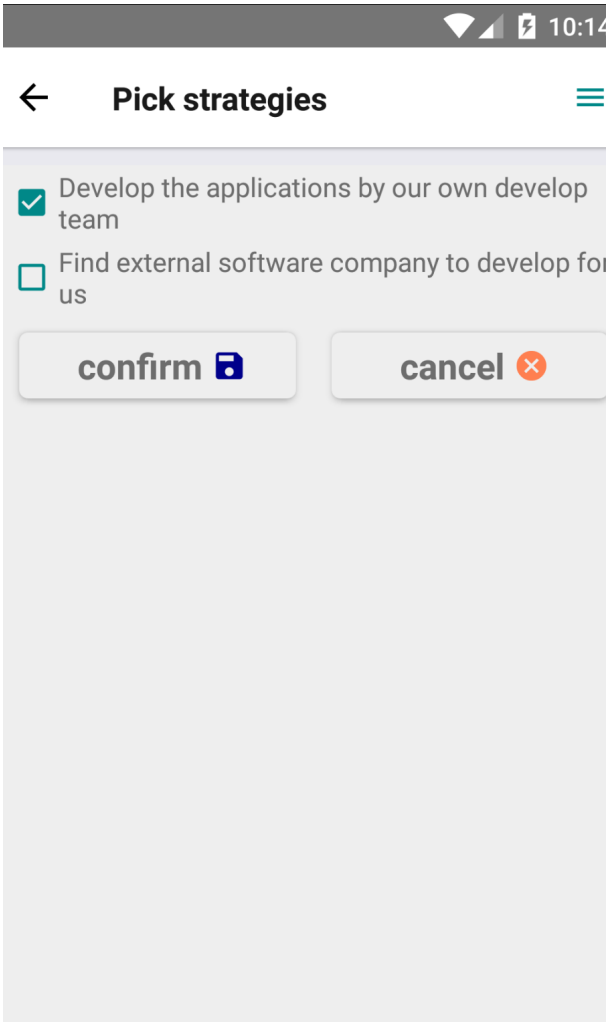


Figure A.11: Selecting Achieving Strategies

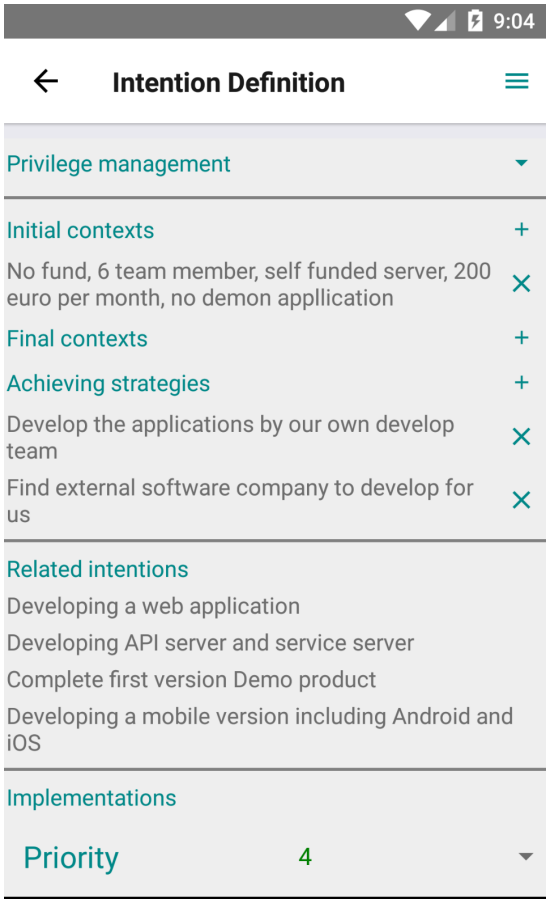


Figure A.12: Related Intentions

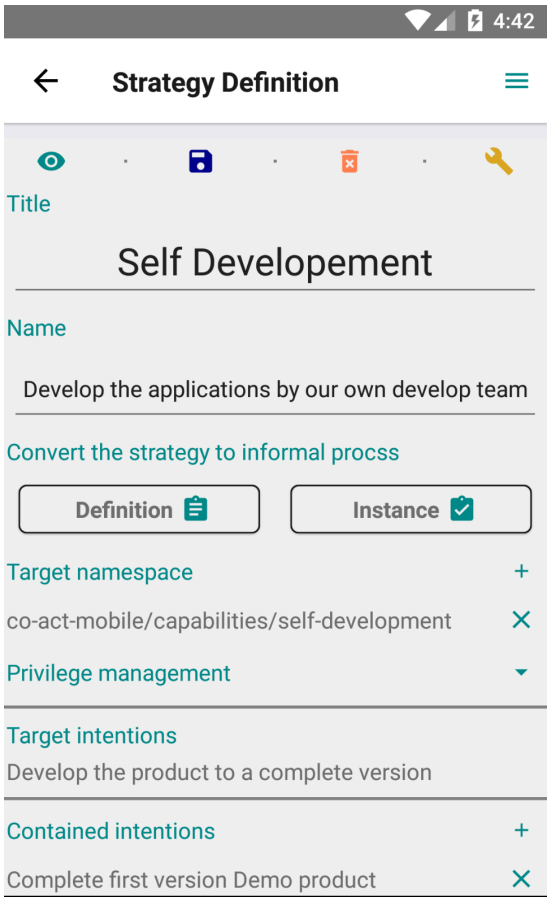


Figure A.13: Target Intention

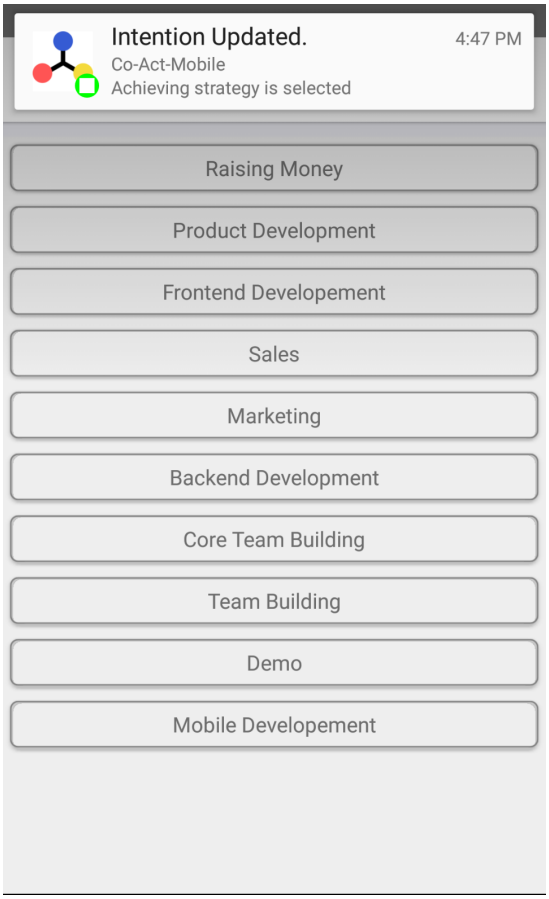


Figure A.14: Push Notification

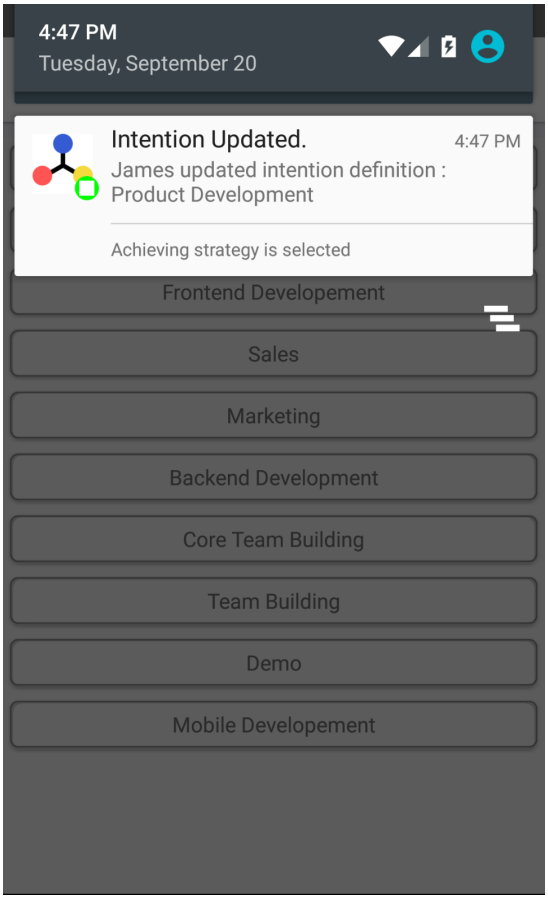


Figure A.15: Notification Detail

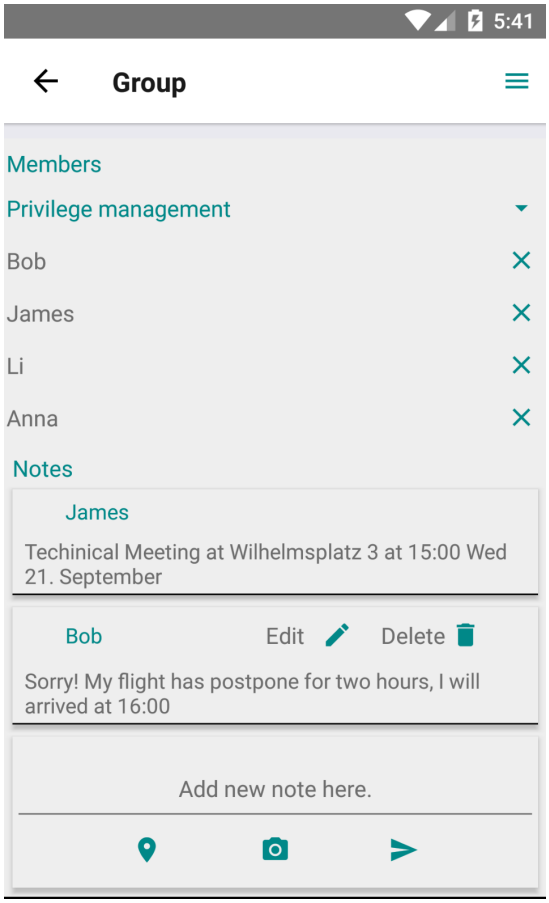


Figure A.16: Acknowledging Flight Delay

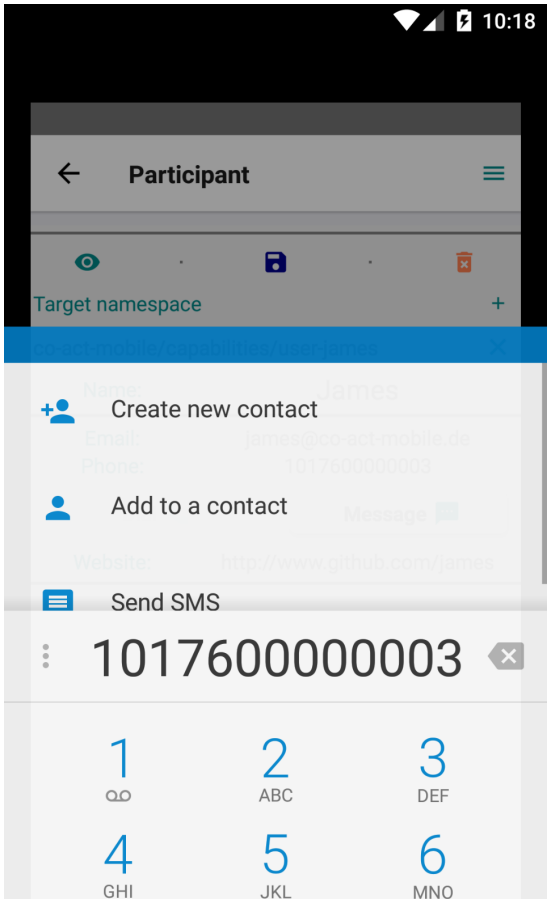


Figure A.17: Dialling Participant from Profile

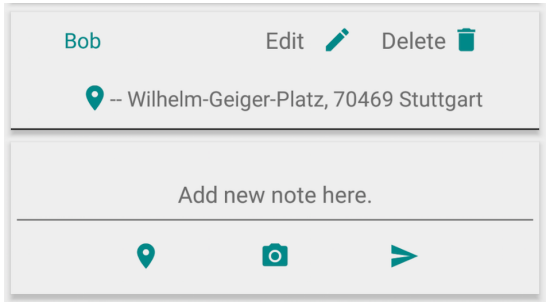


Figure A.18: Note with Location

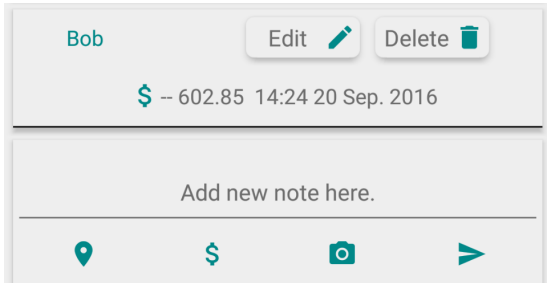


Figure A.19: Getting Bitcoin Price

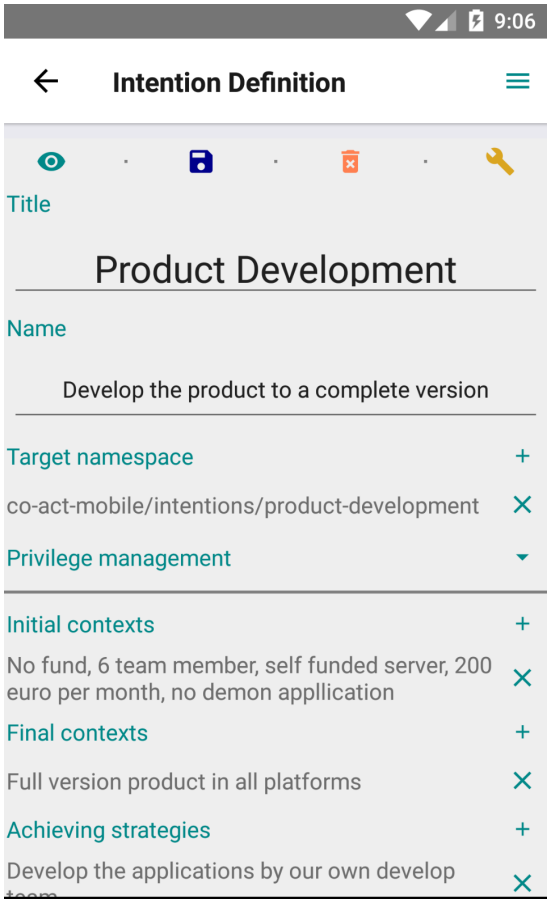


Figure A.20: Final Intention Definition 1

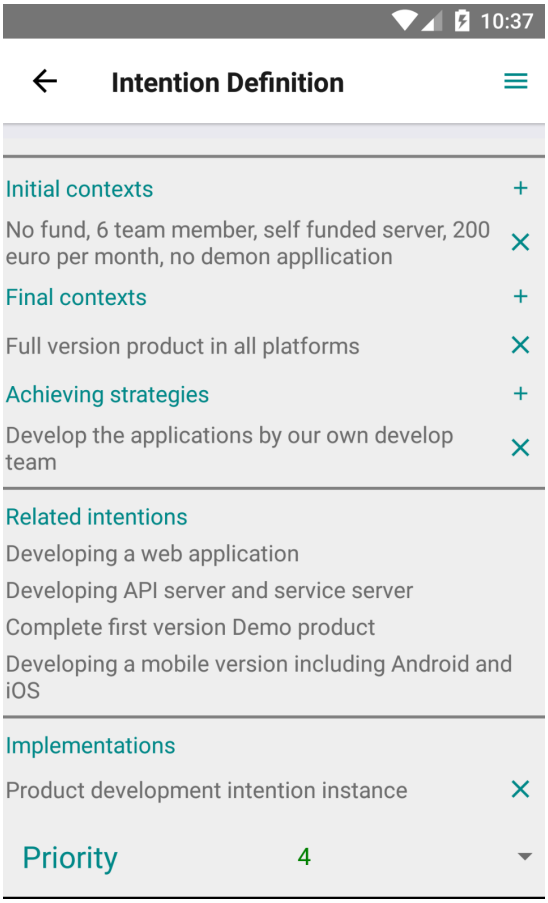


Figure A.21: Final Intention Definition 2

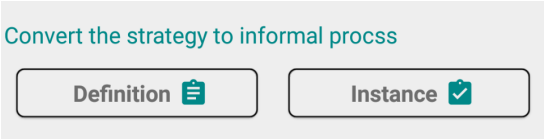


Figure A.22: Convert Strategy

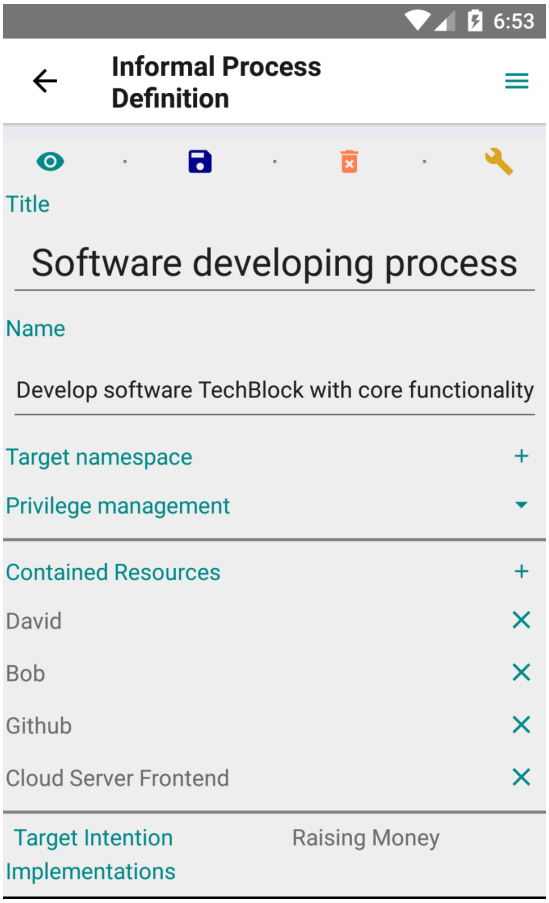


Figure A.23: Generated Informal Process Definition

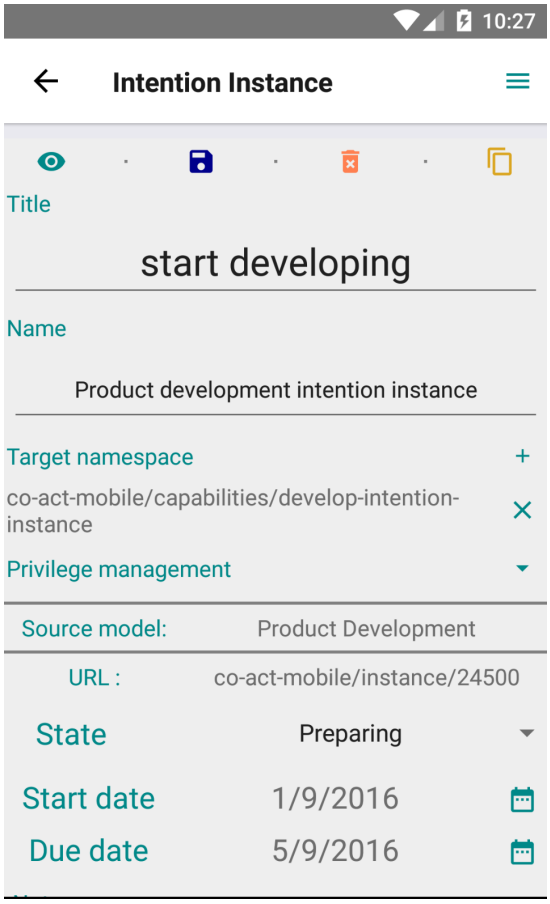


Figure A.24: Intention Instance

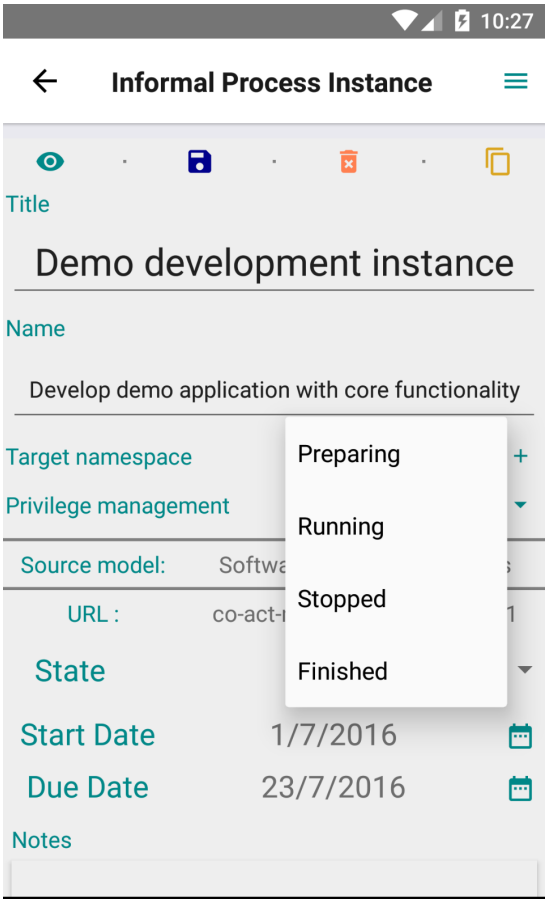


Figure A.25: Informal Process Instance

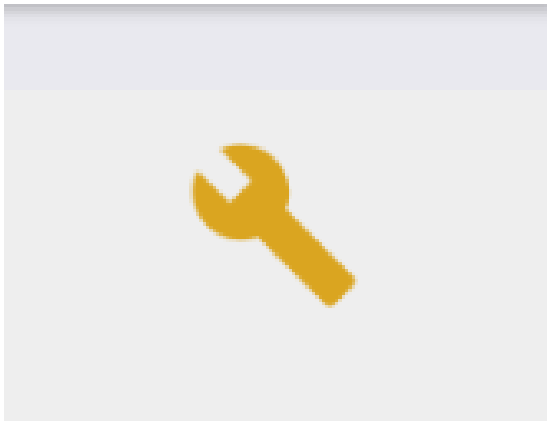


Figure A.26: Initialization Button

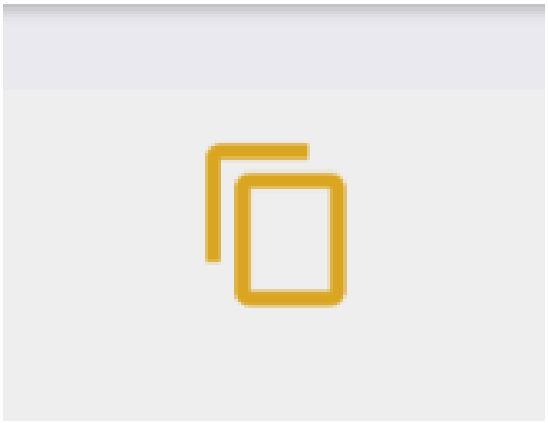


Figure A.27: Extracting Button

Bibliography

[AAD+07] A. Agrawal, M. Amend, M. Das, M. Ford, C. Keller, M. Kloppmann, D. König, F. Leymann, R. Müller, G. Pfau, et al. "Web services human task (ws-humantask), version 1.0." In: *June, available at http://download. boulder. ibm. com/ibmdl/pub/software/dw/specs/ws-bpel4people/WS-HumanTask v1. pdf* (2007) (cit. on p. 25).

[AIM10] L. Atzori, A. Iera, G. Morabito. "The internet of things: A survey." In: *Computer networks* 54.15 (2010), pp. 2787–2805 (cit. on p. 23).

[BBD11] K. M. Bell, D. N. Bleau, J. T. Davey. *Push notification service*. US Patent 8,064,896. Nov. 2011 (cit. on p. 19).

[BBH+13] T. Binz, U. Breitenbücher, F. Haupt, O. Kopp, F. Leymann, A. Nowak, S. Wagner. "OpenTOSCA–a runtime for TOSCA-based cloud applications." In: *International Conference on Service-Oriented Computing*. Springer. 2013, pp. 692–695 (cit. on p. 25).

[BRV00] J. Becker, M. Rosemann, C. Von Uthmann. "Guidelines of business process modeling." In: *Business Process Management*. Springer, 2000, pp. 30–49 (cit. on pp. 15, 17).

[Car99] E. Carmel. *Global software teams: collaborating across borders and time zones*. Prentice Hall PTR, 1999 (cit. on pp. 15, 19).

[CL11] A. Charland, B. Leroux. "Mobile application development: web vs. native." In: *Communications of the ACM* 54.5 (2011), pp. 49–53 (cit. on p. 47).

[Cle11] N. Clevenger. *iPad in the enterprise: Developing and deploying business applications*. John Wiley & Sons, 2011 (cit. on pp. 18, 25).

[DMR15] C. Di Ciccio, A. Marrella, A. Russo. "Knowledge-intensive processes: Characteristics, requirements and analysis of contemporary approaches." In: *Journal on Data Semantics* 4.1 (2015), pp. 29–57 (cit. on p. 17).

[Dru99] P. F. Drucker. "Knowledge-worker productivity: The biggest challenge." In: *California management review* 41.2 (1999), pp. 79–94 (cit. on p. 18).

[DS+90] T. H. Davenport, J. E. Short, et al. "The new industrial engineering: information technology and business process redesign." In: (1990) (cit. on pp. 15, 17).

[DT12] C. Dorn, R. N. Taylor. “Architecture-driven modeling of adaptive collaboration structures in large-scale social web applications.” In: *International Conference on Web Information Systems Engineering*. Springer. 2012, pp. 143–156 (cit. on p. 25).

[GSD12] A. Giessmann, K. Stanoevska-Slabeva, B. De Visser. “Mobile Enterprise Applications–Current State and Future Directions.” In: *System Science (HICSS), 2012 45th Hawaii International Conference on*. IEEE. 2012, pp. 1363–1372 (cit. on p. 27).

[HFD13] R. Harrison, D. Flood, D. Duce. “Usability of mobile applications: literature review and rationale for a new usability model.” In: *Journal of Interaction Science* 1.1 (2013), p. 1 (cit. on p. 26).

[HGKM14a] E. Hoos, C. Gröger, S. Kramer, B. Mitschang. “Improving Business Processes Through Mobile Apps-An Analysis Framework to Identify Value-added App Usage Scenarios.” In: *ICEIS (2)*. 2014, pp. 71–82 (cit. on p. 27).

[HGKM14b] E. Hoos, C. Gröger, S. Kramer, B. Mitschang. “ValueApping: An Analysis Method to Identify Value-Adding Mobile Enterprise Apps in Business Processes.” In: *International Conference on Enterprise Information Systems*. Springer. 2014, pp. 222–243 (cit. on pp. 27, 29).

[HGM15] E. Hoos, C. Gröger, B. Mitschang. “Mobile apps in engineering: a process-driven analysis of business potentials and technical challenges.” In: *Procedia CIRP* 33 (2015), pp. 17–22 (cit. on p. 19).

[Hoo14] E. Hoos. “Design method for developing a Mobile Engineering-Application Middleware (MEAM).” In: *Pervasive Computing and Communications Workshops (PERCOM Workshops), 2014 IEEE International Conference on*. IEEE. 2014, pp. 176–177 (cit. on p. 26).

[Kal16] A. Kalidoss. “Intention-oriented Organizational Modeling - A Top-down Approach.” MA thesis. University of Stuttgart, 2016 (cit. on pp. 21, 22).

[KBBL13] O. Kopp, T. Binz, U. Breitenbücher, F. Leymann. “Winery–a modeling tool for TOSCA-based cloud applications.” In: *International Conference on Service-Oriented Computing*. Springer. 2013, pp. 700–704 (cit. on p. 25).

[LKAA96] S. Long, R. Kooper, G. D. Abowd, C. G. Atkeson. “Rapid prototyping of mobile context-aware applications: The cyberguide case study.” In: *Proceedings of the 2nd annual international conference on Mobile computing and networking*. ACM. 1996, pp. 97–107 (cit. on p. 25).

[LO12] A. C. Lowry, E. A. Onnen. *Push notification delivery system*. US Patent 8,244,822. Aug. 2012 (cit. on p. 25).

[MDS+10] D. Mukherjee, P. Dhoolia, S. Sinha, A. J. Rembert, M. G. Nanda. “From informal process diagrams to formal process models.” In: *International Conference on Business Process Management*. Springer. 2010, pp. 145–161 (cit. on p. 17).

[NEK+05] S. Nurcan, A. Etien, R. Kaabi, I. Zoukar, C. Rolland. “A strategy driven business process modelling approach.” In: *Business Process Management Journal* 11.6 (2005), pp. 628–649 (cit. on p. 20).

[OOM95] J. S. Olson, G. M. Olson, D. K. Meader. “What mix of video and audio is useful for small groups doing remote real-time design work?” In: *Proceedings of the SIGCHI conference on Human factors in computing systems*. ACM Press/Addison-Wesley Publishing Co. 1995, pp. 362–368 (cit. on p. 15).

[SBBL14] C. T. Sungur, T. Binz, U. Breitenbücher, F. Leymann. “Informal process essentials.” In: *Enterprise Distributed Object Computing Conference (EDOC), 2014 IEEE 18th International*. IEEE. 2014, pp. 200–209 (cit. on pp. 17, 20–22, 24).

[SBD+16] C. T. Sungur, U. Breitenbücher, S. Dustdar, O. Kopp, F. Leymann, C. Mayr-Dorn, M. Song, A. Weiß. “Identifying Relevant Resources And Relevant Capabilities of Collaborations – A Case Study.” In: *EDOC Demonstration Track*. accepted for publication. IEEE Computer Society, 2016 (cit. on pp. 15, 17, 18, 20, 21, 24, 26, 30).

[SBLW15] C. T. Sungur, U. Breitenbücher, F. Leymann, J. Wettinger. “Executing informal processes.” In: *Proceedings of the 17th International Conference on Information Integration and Web-based Applications & Services*. ACM. 2015, p. 54 (cit. on pp. 17, 18, 20, 24).

[SCFY96] R. S. Sandhu, E. J. Coynek, H. L. Feinsteink, C. E. Youmank. “Role-based access control models yz.” In: *IEEE computer* 29.2 (1996), pp. 38–47 (cit. on p. 26).

[SDDL15] C. T. Sungur, C. Dorn, S. Dustdar, F. Leymann. “Transforming collaboration structures into deployable informal processes.” In: *International Conference on Web Engineering*. Springer. 2015, pp. 231–250 (cit. on p. 25).

[Shn10] B. Shneiderman. *Designing the user interface: strategies for effective human-computer interaction*. Pearson Education India, 2010 (cit. on pp. 34, 55).

[Sim64] H. A. Simon. “On the concept of organizational goal.” In: *Administrative science quarterly* (1964), pp. 1–22 (cit. on p. 17).

[SKL14] C. T. Sungur, O. Kopp, F. Leymann. “Supporting Informal Processes.” In: *ZEUS*. Citeseer. 2014, pp. 49–56 (cit. on pp. 20, 23).

[Swe88] J. Sweller. “Cognitive load during problem solving: Effects on learning.” In: *Cognitive science* 12.2 (1988), pp. 257–285 (cit. on p. 26).

[VHH+11] R. Vaculin, R. Hull, T. Heath, C. Cochran, A. Nigam, P. Sukaviriya. “Declarative business artifact centric modeling of decision and knowledge intensive business processes.” In: *Enterprise Distributed Object Computing Conference (EDOC), 2011 15th IEEE International*. IEEE. 2011, pp. 151–160 (cit. on p. 17).

[Was10] A. I. Wasserman. “Software engineering issues for mobile application development.” In: *Proceedings of the FSE/SDP workshop on Future of software engineering research*. ACM. 2010, pp. 397–400 (cit. on pp. 25, 26).

[WW05] J.-H. Wu, S.-C. Wang. “What drives mobile commerce? An empirical evaluation of the revised technology acceptance model.” In: *Information & management* 42.5 (2005), pp. 719–729 (cit. on p. 15).

All links were last followed on Sep 27, 2016.

Declaration

I hereby declare that the work presented in this thesis is entirely my own and that I did not use any other sources and references than the listed ones. I have marked all direct or indirect statements from other sources contained therein as quotations. Neither this work nor significant parts of it were part of another examination procedure. I have not published this work in whole or in part before. The electronic copy is consistent with all submitted copies.

---

place, date, signature