What was I meant to do again (Exploration of event boundary on the failure of prospective memory)

Aldy Syahdeini

Master of Science
School of Informatics
University of Edinburgh
2017

Abstract

Prospective memory is remembering to carry out planned action in the future wihtout instructed to do so. The prospective memory failure is the common phenomena in everyday life. The effect of prospective memory failure can be trivial but can also be critical. The topic and the experiment of this research is based on the experiment conducted by Carlson and his team. Different component of prospective memory and what aspect influence people to experience the failure was investigated.

An application to conduct a prospective memory experiment is developed. The application is made flexible so different type of experiment properties can be implemented in the experiment. The experiment on the failure of prospective memory is also conducted. Data were derived from 18 participants who underwent 3 experimental studies. The experiment shows that the prospective memory error happens when people use a smartphone. The amount of intentions is a crucial component of prospective memory and the mental and physical transition influence of prospective memory error.

Acknowledgements

Many thanks to my supervisor Dr. Maria wolters who put her believe on me, and her guidance on the thesis. Special thanks to Catherine crompton for useful discussions, hints, and advice. thanks to Prof. Richard Alan Carlson who give me an further understanding about this topic. Big thanks to all my family and friends who always support me when I am working on my thesis. Lastly, thank you for all the participants who participate on this experiment.

Declaration

I declare that this thesis was composed by myself, that the work contained herein is my own except where explicitly stated otherwise in the text, and that this work has not been submitted for any other degree or professional qualification except as specified.

(*Aldy Syahdeini*)

Table of Contents

Don't worry, it's gonna be good in the end

- Dr. Maria wolters

Chapter 1

Introduction

Every moment of consciousness is a precious and fragile gift.

Steven Pinker

1.1 Prospective Memory error

Have you experienced when you wake up from your bed in the morning, put your glasses on and go to the kitchen to get a glass of milk. But when you are in the kitchen, you forget what you intended to do. This phenomenon is called prospective memories failures.

Prospective memory is the ability in the future to remember to do an action that previously planned without being instructed to do so (Groot et al., 2002). This type of memory is different with retrospective memory which is the memory that we use when we are answering a question in the exam. Retrospective memory involves remembering event, words, and so on from the past typically when deliberating to do so.

Prospective memory failures are common in everyday life, almost 50% of forgetting in our daily routines are due to of prospective memory error (Crovitz and Daniel, 1984). This memory failure can lead to embarrassment such as forget that you had arranged a meeting with your friend and even result in serious injury or death. One example of a horrible case is ”After a change in his usual routine; an adoring father forgets to turn toward the daycare center and instead drove his usual route to work at the university. Several hours later, his infant son, who had been quietly asleep in the back seat, was dead ” (Einstein and McDaniel, 2005). So it is important to have a great understanding about prospective memory error.

But what makes us forget ?. Radvansky and Copeland (2006) and Radvansky et al. (2010) shows that if people make a transition from one event to another, for examples move from one room to another room, they tend to forget more information than if they do not. Cockburn and Smith (1994) Show that stress and anxiety cause us to become absent-minded and thus produce failures of prospective memory. There is also a lot of study about ageing and its relation to prospective memory, one of it is a study conducted by Scullin et al. (2012) found older people tend to make more error than younger people on a prospective memory test.

The purpose of this MSc Dissertation project is to build an application that can use to conduct an experiment about prospective memory error, analyse the effect of multiple intention on a failure of prospective memory and to make a further understanding of what happens during event boundary (e.g., moving to another application inside the smartphone) by tracking the activity of the participant during the prospective memory task. The experiment conducted on this thesis is originally based on studies done by Carlson (what Did I Come here to do ?, Pennsylvania State University 2016).

1.2 Project goals

The main goals of the thesis are to create an application that can be used to other researchers to conduct a prospective memory experiment. The application should able to conduct three type of studies from Carlson's experiment. Three studies are conducted to analyze the influence of multiple intentions (is attentional loads matter?) and event boundaries (event horizon model) on prospective memory error.

1.3 Structure of dissertation

The document is structured as follows

- In the **Literature Review** chapter provides an explanation about the prospective memory, retrospective memory and what influence the prospective memory error. The different element of prospective memory is explained here.
- In the **Experiment and Application Design** chapter provides an information about the architecture and the design of the experiment and the application. How the experiment is conducted and its properties are explained. The main flow and the user design of the application are also provided.

- In the **Implementation** chapter provides information about the technical implementation of the experiment application based on the design and the requirement. This chapter explains how the flow of the application works and how the features is implemented.
- In the **Experiment result and Discussion** chapter provides the result and analysis of the output of the experiment.
- in the **Conclusion and Suggestion** highlight the summary and achievement of the application and experiment. And also giving an opinion about possible future improvement and research.

Chapter 2

Literature Review

2.1 Prospective memory and retrospective memory

The mind is a set of modules.

Steven Pinker

Tasks such as buying milk in a supermarket on the way to work action, turning off the oven and taking a medication are categorized as a prospective memory task. Prospective memory is used constantly in everyday activity (Wilkins and Baddeley, 1978), (Winograd, 1991). There are a lot of definition about prospective memory, but generally a prospective memory is defined as remembering to carry out planned actions at a particular time in the future without being instructed to do so (McDaniel and Einstein, 2007); (Groot et al., 2002). While task such as answering the question on an exam or remembering the person name on the party is categorized as a retrospective memory task. Retrospective memory involves remembering events, words, and so on from the past typically when deliberating trying to do so.

According to Baddeley and Wilkins (1983), it's very hard to differentiate between prospective memory and retrospective memory because there is no clear cut between them, for example, To remember to call your father, you should able to recall his number and how to use the phone, and not call him while he watches a football match. Brandimonte et al. (1996) call this as the retrospective component of a prospective memory task. Cockburn (1995) stated that content of the information is similar to both memory type but the essential difference is prospective memory require memory for intention and the cue for retrieval has to be self-initiated. Guynn et al. (1998) also state that retrospective memory is driven by low information content while retrospective memory is driven by high perceptual information, such as question during an exam.

Furthermore, Remembering only the retrospective memory component of a prospective memory task will not produce successful prospective memory. In fact, numerous prospective memory failures happened because the failure of remembering the prospective memory component (Einstein et al., 1992). Interestingly, the component of retrospective memory sometimes forgotten in a simple prospective memory task, for instance when we walk to the kitchen and sometimes forget what we are intended to do there (Brandimonte et al., 1996).

2.2 Cognitive process of prospective memory

Some researcher believes that prospective memory proceeds through encoding, retention, retrieval, execution and evaluation phase. According to Ellis (1996) In the Encoding phase, the *when*(retrieval criterion), *what*(action to be performed) and *that*(intent or decision to act) are encoded. Then this intention representation must be retained until the opportunity to fulfil the intention occurs. This delayed can vary from a second to a week. Einstein and Mcdaniel (1990) categorize retrieval process into two categories; event-based retrieval and time-based retrieval. On the event based retrieval, the retrieval happens if there is a particular event or physical stimulus that associated with the intention. for example telling a message when you meet your college. On the other hand, time-based retrieval require execution of action after a certain time (Ellis, 1996); (Mcgann et al., 2002). Therefore, successful prospective remembering can be described as a process that supports the actualization of delayed attention and the associated action, and it is strongly associated with control or coordination of future action (Ellis, 1996).

2.3 Prospective memory error

Prospective memory error is defined as a failure to do a planned action at some point or a particular event in the future. Kliegel and Martin (2003) state that prospective memory failure is the most frequent memory failure in everyday life. The ability to remember the planned action is a critical factor in human functioning. The consequence of a failure of prospective memory can be trivial, for example forgetting to buy some milk on the way home from work. But it can also have severe consequences, for example the doctor forget to took the scalpel from his patient after an operation. In fact, Shorrock (2005) reported that 38% of accidents on the traffic controllers in the UK was due to memory error involves the failure of prospective memory.

Many researchers have different view on the prospective memory error and what cause it to happen.

Kvavilashvili and Ellis (1996) try to differentiate a various kind of memory error with a prospective memory error. They claim that *action-slip*(Heckhausen and Beckmann, 1990), *actions-not-as-planned* (Reason, 1979) and *absent-minded error* (Cohen and Conway, 2008) should not be considered as a prospective memory error. These errors happen because the failure that occurs during the execution or performance of the intended action, for example in absent minded error people lose the context of an intention and carry out an unintended action instead of the intended one. In contrast, prospective memory is focused on the failure to retrieve intended action. While Guez and Naveh-Benjamin (2013) argue that these type of error should be considered as part of prospective memory error because prospective memory contains some element of retrospective memory such as the context of intention. Moreover, Reason (1985) explained further on how the element of memory; context, intention and attention influence prospective memory error. In addition, Cockburn and Smith (1994) argued that stress and anxiety make a person to experience absent minded error hence make a prospective memory error, and Scullin et al. (2012) found older people tend to make more error than younger people on a prospective memory test.

2.4 Prospective memory and intention

Because prospective memory refers to remembering intentions so it would be better to have a good understanding of intention first. For example to understand the nature of intention and its phenomena, the category of intention and how it related to everyday activities and what happens to intention during prospective memory error. The explanation of these question maybe gives us more understanding about the correlation between intention and prospective memory error.

Kvavilashvili and Ellis (1996), Gauld and Shotter (1977) define an intention as a person's readiness to act in a certain way in the future. What has to be done and when to be done should be defined clearly. Searle (1983) distinguished intention into two types, prior-intention and intention-in-action. A prior intention is an intention that is defined prior to action, while intention-in-action is a spontaneous action, for example going to the toilet when you need to urinate. A prior intention always occurred as a result of conscious decision to act in a certain way (Heckhausen and Kuhl, 1985). Furthermore, Gauld and Shotter (1977) categorized prior intention into two categories, delayed intentions and immediate intention.The delayed intention is a postponed intention that will be executed at some point in the future, and when a person begins to carry out their prior intention immediately after a decision has been made or after they see a particular cue for the intention.

The difficulty of retrieval of the delayed intention make persons miss the prearranged moment or cues, and this makes people fail to remember. Even though people able to retrieve the delayed intention, but when the intention is initiated and transformed into an immediate intention, people can still lose their intention and prospective memory error occurs. Furthermore, Reason (1985) explain how a change in the intention make people experience memory error by categorizing two phenomena called *detached intention* and *lost intention*.

2.4.0.1 Detached intention

Detached intention happen if the original content detached from the intention. it will then get replaced or misaplied to another content apart from its origin. For example, the case when a person switches off the television instead of the oven. (Reason, 1985) explaned that this phenomena happened because the intention is not framed completely. This premature intention happen probably because a person's attention is focused on other things (this will be explained further on the attention section). Another explanation is the intention is replaced because it it do not has a sufficient level of retaintion even though the intention is framed completely. Another explanation is an existence of intention that has similar content and triger from same object which similar kind of action is appropriate (Reason, 1985).

2.4.0.2 Lost of intention

In contrast with detached intentions that happened because of partial failure of the intention and retention system, lost intention is a complete failure at one or more of the stage of formulation, encoding, storage, or retrieval of the intention. One typical case is when an intention is lost during the retrieval phase, for instance when a person walks into a room and become aware that he/she can't recall the original intention of the activity (Reason, 1985).

2.5 Prospective memory and attention

When we accidentally put our phone in the fridge instead of our food or when we pour the second kettle of water into a freshly made coffee. These slips of action frequently occur as the result of misdirected or diminished attention (Reason, 1985). There is a minimum degree of attentional involvement is necessary to ensure the right execution of the sequence of attentions, and to avoid someone make a mistake due to some kind of attentional failure.

Reason (1985) define attentions as the gatekeeper of consciousness. This definition marks an important role of attention and consciousness in the performance of delayed intention on prospective memory. A person must be conscious of the plan to perform an action. To be conscious about it, the plan should be the focus of attention. The attention should be kept at the encoding phase when the action is planned and at retrieval when the action is performed.

But error can also occur when a person is putting too much attention on the ongoing activity, for example, running down the stair two at a time, this should be an automatic activity but when a person does it with too much attention, then it can be very disruptive.

Moreover, dividing attention is also assumed to reduce the contribution of a controlled process, thereby reducing performance on a memory test that involves conscious recollection (Jacoby et al., 1989). Some previous study also shows that there was a substantial reduction in prospective memory performance when attention is divided (McDaniel et al., 1998) (Guez and Naveh-Benjamin, 2013).

2.6 Prospective memory error at event boundaries

We walk to the park, read a book, watch a movie and do numerous things, one after another. These stream of actions consist of events. How we split up these stream of action into events and stored them into memory influence how we think and what to remember. Memory and cognition are heavily influenced by an event and how a person structures them (Radvansky, 2012). Radvansky et al. (2011) introduce an event model which is a mental model that captures the content and structure of an event that people experience.

Radvansky (2012) also suggest that when persons make a cognitive transition from one event to another, they will experience an event boundary. Such transitions can be a change in location, a causal break, the introduction of a new activity, and so on, as long as they involve a shift from one event to another. On some condition, event boundaries can disrupt memory. When people experience event boundaries, they mentally update their event model. Radvansky et al. (2010) investigate about this phenomena in the reading experiment and shows that the updating effect of a mental model increases the reading time of a sentence. The increase of time reflects increase on cognitive effort need for the updating.

Furthermore, Radvansky et al. (2010) found that people forget more information if they pass through the doorway to move from one location to another. This effect is similar to the result from other research in text comprehension that shows that shift in location decline memory performance (Curiel and Radvansky, 2002); (Haenggi et al., 1995); (Radvansky et al., 2010); (Radvansky et al., 2003). Moreover, that study also showed that people were more likely to forget when they passed through two doorways.

Kurby and Zacks (2008) and Swallow et al. (2009) proposed event segmentation theory which explains the correlation between memory and event. The theory stated that during the experience of an event, when event boundaries are identified, people segmented information into separate event models and then stored it into memory.

All these previous research result in event horizon model proposed by Radvansky (2012) This model also supports an event segmentation theory. The model explained that when an event is segmented and stored as event model, it declines in availability and become deactivated. And as person experience event boundaries, a new event model is created in working memory. The active event model that is currently in the working memory is foregrounded which make it easier to retrieve, and an available processing capacity is directed to it.

The presentation of a memory cue causes both models that contain target information to be activated this result on competition and interference, which slows down response times and increases error rates. This is why returning to a previous room does not improve memory for objects that were encountered there, and why passing through two doorways makes memory even worse than does passing through one (Radvansky et al., 2011).

2.7 The Experiment

The experiment on this thesis is based on the experiment conducted by Lisa. M. Stevenson and Richard A. Carlson from Pennsylvania State University conducted an experiment on the failure of prospective memory. Each participant answer trivia questions. On each question, an embedded link is presented, and the participant was instructed to find the answer on the web page. Subsequently, the participant is asked questions to assess their prospective memory. The experiment conducted three studies, the studies are explained more on section 3.1.1. Carlson found that a failure of prospective memory happened when a person uses a smartphone. They found that the amount of intention is an important factor of prospective memory (intentional loads matter). Also, they found that there is no improvement or decrement on the failure of prospective memory after the transition in locations.

Carlson implemented the experiment by using Qualtrics, a web-based questionnaire administration tool. And the participant used their smartphone browser to access the website. While we implemented the experiment by using android application. While Carlson mostly assessed the participant failure of prospective memory by using a question at the end of the experiment. We were also asked similar questions, but we also track some variables of participant's activity during the experiment.

We track the frequency of the participant when they forgot the question and decided to see it again, this is useful to analyze what factors made the participant experienced failure of prospective memory. We track how long the participant spent on finding the answer to understand analyze the retention and retrieval of the intention. We also track how long they spent writing the answer to analyze how they retrieve the content of the intention. All the tracked variables are explained more detail on section 3.2.6 and the implementation in section 4.4.

Notification were also shown to the participant during the experiment. The aim of the notification is to lower the level of attention of the participant. The output of the tracked variable and the occurrence of the notification were analyzed to see the effect of notification on a failure of prospective memory. The design of the notification is explained further in section 3.2.6, and the notification mechanism in section 4.3.

In addition, based an idea of Ellis (1996), some may argue that the type of intention on the experiment is not delayed intention thus it cannot be associated with a prospective memory error. But this view is refuted by Guez and Naveh-Benjamin (2013). According to Carlson, there should be a temporal gap Between forming the intention and the opportunity to carry it out. Typically, of course, part of that interval is filled by some other task. In the case of the phenomenon we are trying to capture, that other task is simply moving (physically or on the phone/computer) to the setting that allows the intention to be carried out.

Chapter 3

Experiment and Application Design

> If the brain was simple enough to be understood - we would be too simple to understand it!.
>
> *Minsky M.A.*

This chapter describes the design of the experiment framework, both the experiment and the application side. The application will be explained from the system design point-of-view and the user experience perspective. The application high-level decision and work flow are explained using a flow chart and a class diagram.

3.1 Experiment design

3.1.1 Studies

The experiment is based on Carlson experiment. We modify several properties and add some features to the experiment; Carlson experiment ask eight questions on each study, we asked ten questions. In Carlson experiment, the category of the question consists of movies and TV, geography, or Penn State trivia. But we only used the movies category. Moreover, In the third study, Carlson instructed the participant to move within a room, between rooms or stay seated. While we instructed the participant to move from room to a corridor, and in reverse.

In Carlson experiment, they assessed the failure of prospective memory by asking the participant the first question from Table 3.2. We also asked the question, but we also track some variable during the experiment to help us make better understanding of the phenomena. The list of tracked variables is explained in the section 3.2.6.

A number of notifications were also shown during the experiment, this notification is used as a distraction to lower the level of attention of the participant. The notification design is explained in the section 3.2.5 and the mechanism is explained in section 4.3.

Our experiment result is used to confirm the result of Carlson experiment . By analyzing the participant activity during the experiment, we hope to makes further understanding of what happened during the failure of prospective memory. The experiment consists of three studies;

3.1.1.1 Study 1

The aim of this study was to examine if prospective memory failure happens if a person uses a smartphone. 63 participants participated in the study. Two question presented at a time. Participant used a phone in the lab or the empty room. Carlson found that 75% participant experience prospective memory. This shows that failure of prospective memory happened when a person uses a phone.

3.1.1.2 Study 2

The aim of this study was to examine if increasing number of intentions influence failure of prospective memory (is intentional loads matter?) The number of intention presented as a number of questions at a time. 30 participants participated in the study. One or two questions presented at a time. Participant used a phone in the lab or the empty room. Carlson found that failure of prospective memory happened more frequently if the participant were presented with two questions at a time, thus the intentional loads matter.

3.1.1.3 Study 3

The aim of this study was to examine if event boundary influence memory. This is based on the event horizon hypothesis (Radvansky et al., 2010) and event segmentation theory (Kurby and Zacks, 2008). One question is asked at a time. Every time the participant see the question, they are instructed to move between rooms. Carlson found that there is no improvement or decrement on the failure of prospective memory after the transition in locations.

3.1.2 Participant

21 Participants participated in the study. Three participants were pilot tester. The remaining 18 participants were students from the University of Edinburgh. The participant consisted of 4 women and 13 men. The age of participants was between 21 to 27 years old. Four participants participated in the first study. 11 participants participated in the second study, and the remaining three participants participated in the third study.

3.1.3 Procedure

The experiment is conducted as a form of quiz where a number of questions are presented to the participant and the participant need to look the answer on the answer page. More detailed about the experiment can be seen on experiment design section 3.1. Three studies were conducted. On each study, ten questions with the same category were presented to the participant. The study was conducted in a silent lab room in a forest hill lab and a meeting room on the library. The room was kept empty and quite to lower the level distraction. On the third study, every time the participant look at the question they were instructed to move from room to a corridor.

3.1.4 Question

During the experiment, ten questions are used. The questions are designed as simple as possible so it does not require the participant to remember long context of the question. The questions and its answer are listed in Table 3.1

3.1.5 Demographic question

The participant is required to answer demographic questions at the end of the experiment. The list of the demographic questions can be seen on table 3.2

3.1.6 Input Data

The input data used in this experiment can be seen on the github application repository.

3.1.7 Consent form

A consent form is used to get the approval of the participant and to ensure the participant understand the experiment. The participant will need to sign the document. The

No	Question	Answer Link	Answer
1	What is the original name of the titanic movie ?	https://www.simplemost.com/15-fun-facts-probably-didnt-know-titanic/	Planet
2	In the movie "Lord of the Rings", How tall is Gandalf ?	https://www.phactual.com/14-fun-facts-about-the-lord-of-the-rings-the-fellowship-of-the-ring/	Seven foot
3	How many actors played both in Game of Thrones and in the Harry Potter movies ?	http://screenrant.com/best-facts-game-of-thrones-trivia/	9
4	What is the meaning of Dumbledore in the Harry Potter movies ?	http://www.teenvogue.com/gallery/harry-potter-facts	Bumbelbee
5	How many years has How i met your mother been filmed ?	https://www.phactual.com/10-fun-facts-about-how-i-met-your-mother/	9 years
6	How many baloons are attached to carls house in the "UP" movie ?	https://filmschoolrejects.com/10-fun-facts-about-pixars-up-1749a61575ca/	10,297
7	Where does marvel get the idea of the black spiderman suit ?	http://screenrant.com/best-marvel-facts-trivia-movies-tv-comics-superheroes/	Fan or Randy
8	What is the most expensive movie of all time ?	https://www.factretriever.com/hollywood-movies-facts	Avatar
9	How many academy awards has the movie "UP" been nominated to ?	http://www.imdb.com/title/tt1049413/trivia	two
10	What is the most watched episode on the show How i met your mother ?	https://ritely.com/how-i-met-your-mother-trivia/	The finale episode

Table 3.1: The questions used in the experiment

document can be seen on figure A.1

3.1.8 Participant information sheet

A participant information is given to the participant before the experiment is conducted. It has the information about the experiment and the protection of the experiment's data. The participant information sheet can be seen on figure A.2

3.2 Application Design

The framework is originally based on the web application built by Carlson and his team. Instead of using a web based application our experiment used an android application installed on a smartphone. The experiment framework is made extendable, dynamic and modifiable so that other researcher can design and conduct various type of experiment using a high number of samples. The Experiment framework consists of an android application and a web server application. The relation between these two components can be seen in figure 3.1. The experiment application also can add notifications and track variables during the experiment.

A researcher will need to start the web server and use it to upload the input file. On the android application, the researcher can set the experiment properties for example which experiment to conduct or the name of current participant etc. So the researcher can conduct multiple experiments with multiple participants without uploading the input file again . Then, the experiment is conducted and the application tracks a group of

No	Question	Answer options
1	Often people go into a room to do something. Though they know they intended to do something,they lose track of what they wanted to do. This same sort of thing can happen when using a smart phone, as well. During the study, you may have clicked on a link, gone to the website, and then forgot what you intended to look up. Did that happen to you at all during this study?	Yes, No
2	During this study, did you ever look up an answer, then forget the answer before you were able to type it in?	Yes, No
3	During the course of this study, how many cell phone notifications did you receive ?’	0,1,2,3 or more
4	How many notifications did you decide to click ?	0,1,2,3 or more
5	As you were looking up information, did you ever follow a link you didnt́ need to follow, just out of interest ?	Yes,No
6	During the study, did you read about or learn any new facts that were not answers to questions we asked ?	Yes, No
7	How old are you ?	-
8	What is your gender ?	-
9	What country are you come from ?	-
10	Is English your native language	Yes,No
11	What kind of phone do you normally use ?	non-smartphone, iphone, android, other
12	How difficult did you find the smartphone in this study ?	Very Hard, Hard, Average, Easy, Very Easy
13	How frequently do you use a smartphone ?	Dont́ own one, Daily, Weekly, Monthly

Table 3.2: The demographic questions used in the experiment

variables. After finishing the experiment the output data inside the android application will be sent to the web server which will be compiled to a JSON output file.

3.2.1 Requirement

Table 3.3 below shows the list of all requirements of the application and its description.

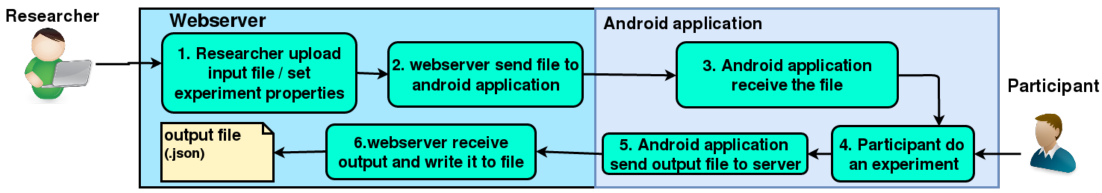

Figure 3.1: Flow of the experiment framework

Requirement List		
No	Name	Description
1	Upload Input	Researcher upload a JSON input file to the application
2	Download Output	Researcher download a JSON output file of the experiment
3	Insert multiple categories	Researcher insert multiple categories
4	Insert questions	On each category researcher insert multiple questions
5	Set number of presented question	Researcher set how many question will be presented on each quiz phase
6	Set presented question behavior	Researcher set whether the number of presented question will be random each phase
7	Insert post question	Researcher insert the question that will be asked after the experiment, e.g demographic question
8	Set experiment properties	Researcher able to set extra experiment's properties apart from input file.
9	Insert notification	Researcher insert notifiaction and information on what is its content, when it will appear
9	See the questions	The participant can see the questions
10	Show answer link and answer page	The participant can see and able to click the answer links
11	Fill the answer	The participant can write an answer
12	Show notification	The application can show the notification
13	Track variables	The application can track defined variables

Table 3.3: List of requirements

3.2.2 Input and Output

The researcher needs to upload the input file that consists of all the experiment properties. After the experiment finishes, the result can be downloaded as a JSON file. JSON (Java Script object notation) is used as an input and output format because it is very easy for a human to read and write, also for the machine to parse and generate. Most of the current programming language and analysis software support JSON format (jso, 2017). The JSON format consists of key and value pairs, on many programming languages it is similar to dictionary, table or struct. This input file will then be uploaded and compiled to the android application. Here is a simple example of the JSON format. To keep the data anonymous the researcher can set the participant id as empty, set parameter of experiment is explained on 4.2.3. The application will generates a uique ID automaticaly that can be used to identify the participant data.

```
{
    name:"John",
    age:21,
    hobby:"swimming"
}
```

Table 3.4 shows all the field for the input and its description. The output of the application will be a JSON file that consists of the experiment result which consist of the answer to the all the questions and tracked variables.

3.2.3 Application Entities

The input file that the researcher uploaded will be generated to an object. The architecture of the object can be seen in figure 3.2. Each box represents an object that consists of properties and methods.

The biggest object is a *study* object, this object is acted as a container for other objects. The *study* object holds another object and controls the flow of the experiment. The arrow in figure 3.2 represent which *experiment*, *category*, *questions* and *notification* will be used in the experiment. The *study* object also acts as a tracker which will tracks variables during the experiment.

The *experiment* object consists of properties on how the experiment will works, e.g experiment name, a number of question will be asked, and how the question will be presented. And each *category* objects consist of *questions* objects.

The researcher is able to choose which *experiment* will be used and which *notification* will be shown. While the participant can choose which *category* they want to answer. This selected *experiment* and *category* objects will be linked by *study* object and compiled as *active category*,*active question* and *active experiment*.

3.2.4 Application flow and properties

The application has general properties described in table 3.5 which will be used to identify the status of the experiment. These also used to decide which notifications to show and what variables to track.

Figure 3.3 shows the flow chart of the quiz experiment, and how the application updated. Figure 3.4 shows the front end the application when the participant does the quiz experiment.

To make it easier for the reader the experiment's flow is divided into four stages;

- **Initialization** : firstly a *phase* variable is initialized. The application then confirm if the experiment is active by ensuring that there are still questions need to

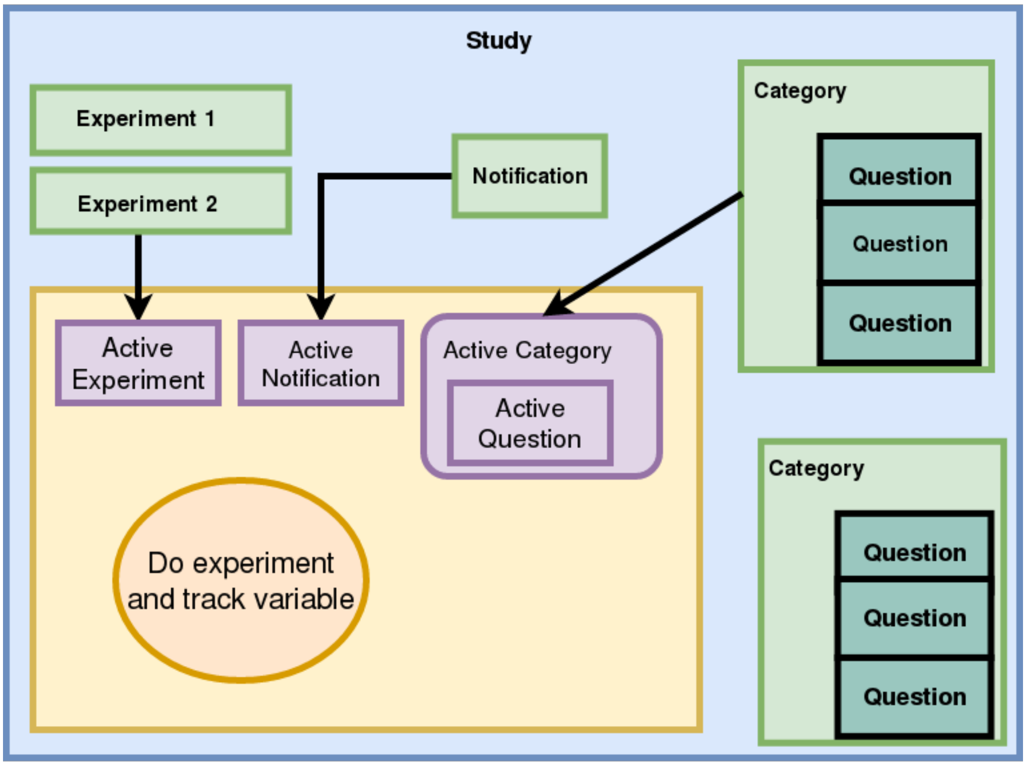

Figure 3.2: Structure of the object inside the application

be asked. The quiz is finished if all the question has been asked.

- **Question activity** : The number of presented question is changed (randomly or constant). Then questions are picked from the *active category* and put to *active question*. Then the *active questions* are presented to the participant.
- **Answer activity** : Thirdly, the links for the answer page are presented to the participant. The participant then click the links and find the answer inside the answer page. The participant is able to see the question again or decide to an-

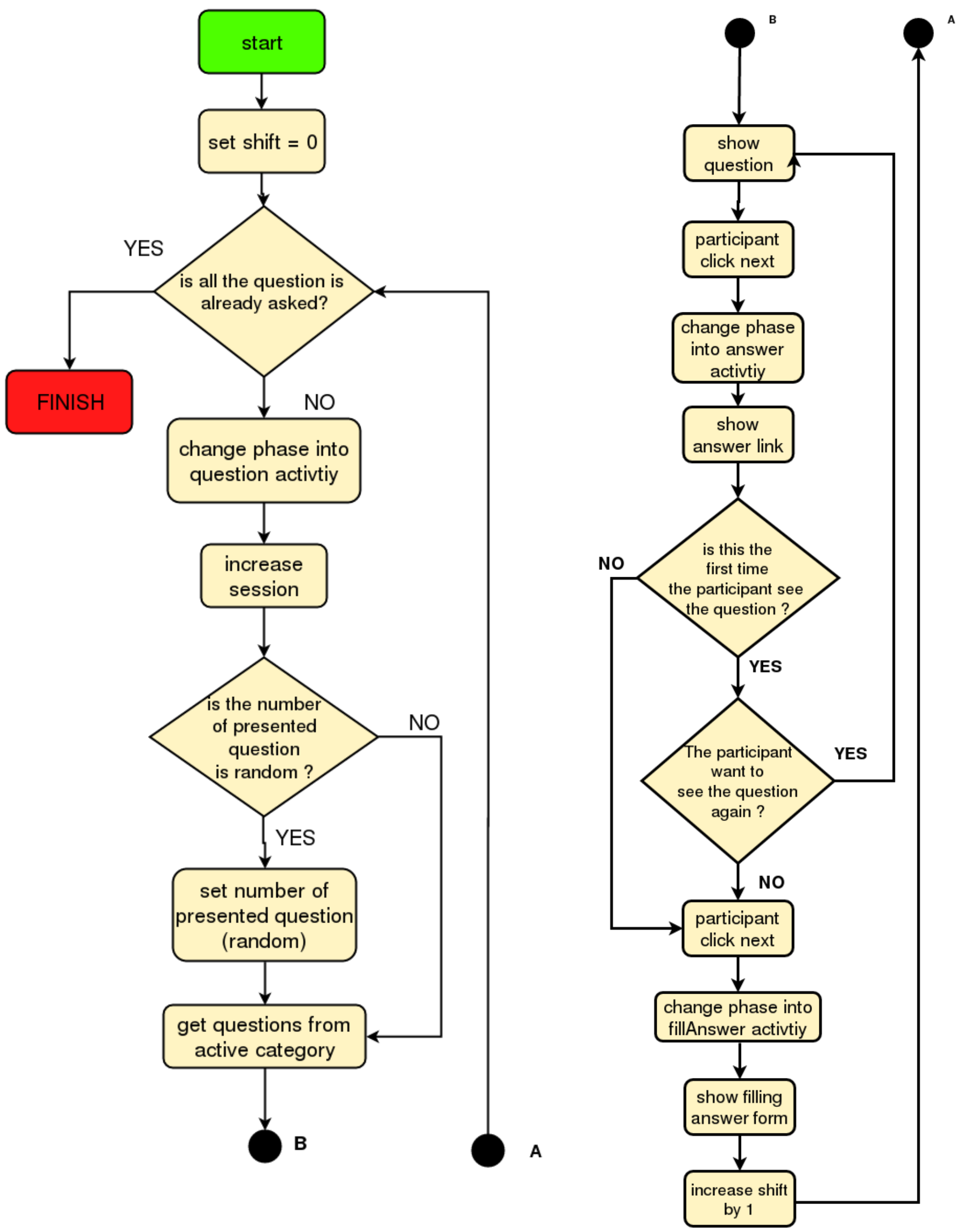

Figure 3.3: Quiz flowchart

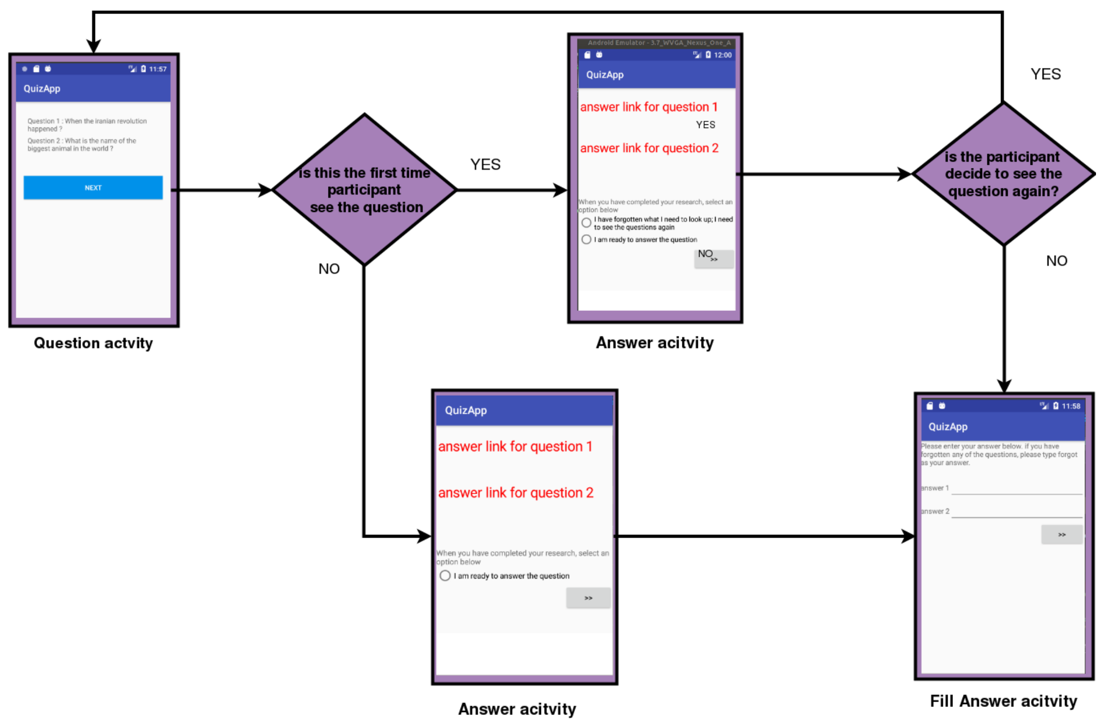

Figure 3.4: Front end of quiz activity flow

swer. The participant is only allowed to see the question again one time.

- **fill answer activity** Lastly, the participant need to write the answer to the question on the text box. After that, the *phase* variable is increased and the application will repeat the quiz again until it finished.

3.2.5 Notification Design

During the quiz experiment, the notification will be shown to the participant. Notification will be shown as a pop-up box as seen in figure 3.5. When the notification is shown to the screen, the phone will vibrate and produce a sound.

If the participant clicks the notification then the application will be minimized and the android phone will be directed to another application. After that, the user can click the application icon to get back to the experiment application.

Based on this design, the *notification* should have the properties listed on table 3.6.

3.2.6 Tracked Variable

During the experiment the application track variables. Table 3.7 consist of all the variables that the application track. Some of the variables have **lb** in front of their name, this means that variable is tracked during the lookback process. A process when the participant looks the question again for the second time.

Figure 3.5: Flow of the notification

Input			
No	Name	Type	Description
1	Study.PreText	String	Html string that will be shown at first on the experiment
2	Study.PostText	String	Html string that will be shown after the pretext
3	Study.Name	String	The name of the study
4	Study.Id	String	The Id of the study
5	Experiment.Name	String	The name of the experiment
6	Experiment.NumQuestion	Integer	The amount of questions to be presented on each quiz phase
7	Experiment.MaxPresentedQuestion	Integer	The maximum number of presented question if it change randomly on each phase
8	Experiment.Random PresentedQuestion	Boolean	Whether the number of presented question will be change randomly on each phase
9	Category.Id	String	Id of the category
10	Category.Name	String	The name of the category
11	Category.TotalQuestion	Integer	The total size of the question on the category
12	Category.QuestionOrder	String	The order of how the question will be pulled from the list of questions. ”LINEAR” it will be pulled based on the input order, ”RANDOM” it will be pulled randomly
13	Category.Question.Id	String	The unique Id of the question
14	Category.Question.Text	String	The question text
15	Category.Question.linkAnswer	String	the URL link of answer
16	Category.Question.Answer	String	the answer of the question
17	Notification.App	String	What application the phone will open if the participant click the notification
18	Notification.shift	Int	The number of phase when the notification should be shown. This will be explained more on the Notification section
19	Notification.Phase	String	The activies name when the notification should be appeared
20	Notification.TimeToShow	Integer	how millisecond the application should wait before showing the notification
21	Notification.Url	String	what url or id the application will open if the participant clicked the notification
22	Notification.TitleText	String	The title of the notification
23	Notification.MsgText	String	The text of the notification

Table 3.4: Explanation of the entities inside the input file

No	Properties name	Type	Function
1	*Shift*	Integer	Used to count how many question-answer had been presented.
2	*Phase*	String	What the name of activity that is currently active on the application.
4	*Active Category*	Category	Contain category used in the experiment, and it contains list of questions.
5	*Active Experiment*	Experiment	Contains the properties of the current experiment.
6	*Number of presented question*	Integer	How many questions are presented at a time (*shift*).
7	*Active Question*	List of Question	List of the question that is presented to the participant.

Table 3.5: List of general properties of the experiment

Variable	Function
shift	This property is to decide on which shitft the notification will be shown, it will be compared to shift variable in experiment properties
phase	This to decide on which activity (question, answer and fillanswer) the notification will be shown.
app	An application the notification will open; instagram, twitter, facebook or website.
timeToshow	Time (in millisecond) notificationwill the notifiaction need to wait before presented.
notification text	The message text inside the pop-up box

Table 3.6: The properties of notification object

Tracked variables list			
No	Variable's name	Type	Description
1	TTLQ	Long	total time (in millisecond) when the participant see the question until they click next button
2	lb_TTLQ	Long	similar with TTLQ, but after lookback
3	LookBack	Boolean	**True** if the participant decide to look at the question again, **false** otherwise.
4	TTLB	Long	total time (in millisecond) when the participant see the answer links until they click the next button (to look the question again or answer the question)
5	lb_TTLB	Long	Similar with TTLB, but after lookback
6	visited_links	List of String	The list of links clicked/visited by the participant after clicking the answer links
7	time_visited_links	List of Long	List of the total time (in millisecond) the participant spent on each asnwer page
8	lb_visited_links	List of String	Similar to visited_links, but after lookback
9	lb_time_visited_links	List of Long	Similar with time_visited_links, but after lookback
10	TTLA	Long	Total time (in millisecond) the participant spent writing the answer
11	TTLFA	Long	Total time (in milisecond) the participant write the answer on the text box
12	num_notif	Integer	how many notification is shown during the a question
13	TTLN	Long	Total time (in millisecond) it tooks the participant after clicking the notification to back to experiment application

Table 3.7: List of tracked variable

Chapter 4

Implementation

The brain is wider than the sky.

Emily Dickinson

This chapter discussed the technical implementation of the system based on the design discussed before. Section 4.1 provides the technical information about the main entity as classes that will be used in this study, including its variables and methods. Section 4.2 provides complete information about the work flow of the application. Section 4.3 provides an information about the mechanism of notification. Lastly, section 4.4 provides an information about the tracker and its procedure.

Flask framework and python programming language is used to develop the webserver. Java programming language and Android SDK is used to develop android application. The source code and the input file of the experiment are put into a github repository at https://github.com/syahdeini/thesis.

4.1 Entities relationship

All the entities discussed in the design chapter will be represented as a class which consists of variables and methods. The relationship between classes can be seen in the class diagram in figure 4.1. The box presents the class. The upper part of the box consists of the variable name and its type. The plus and minus sign before the variable name present the scope of the variable. Minus (-) means private and plus (+) means public. The bottom part of the box consists of the class methods and the type of its output. Furthermore, the arrow presents the variable relationship whether it can be one-to-one (1..1) or one-to-many(1..*) relationship. The arrow also present that a class extend other class which means it has same properties and methods.

4.1.1 Study Class

Study class is the class that acts as a central container for other entities of the application, and it controls the flow of the main function of the experiment. Some of its properties are defined from an input file; *preText, postText, researcher, studyName, studyId*. As seen in the class diagram 4.1. The study class contains *experiments* variable which consists of a list of experiment objects. *Categories* variable which consists of a list of category objects, and *notifs* variable that consists of a list of notification objects.

As explained in the design section, the experiment application need to initialize following variables before conducting the experiment; *ActiveExp* (active experiment) variable is experiment that will be conducted, and *activeCatg* (active category) is the category that will be used to pick questions. *ActiveQuest* (active question) variable present what questions are currently asked. The question inside *activeQuest* variable is picked from a list of questions inside the active category (*activeCatg*) variable. *ActiveExp* is set by the researcher on the setting experiment window, and the *activeCatg* is chosen by the participant during the experiment

Moreover, the class also contains *activeNotif* (active notification) variable. This variable contains a list of notifications that have been presented to the participant. The notification object is picked from the *notifications* variable that contains all the notification.

4.1.2 Experiment Class

The Experiment class contains the *experiment variables* that is used to define the behaviour of the experiment. The variables are explained in the table 4.1. Every variables inside this class except *numPresentedQuestion* is determined from the input file. *numPresentedQuestion* is a variable that controls how many questions is presented to the participant each time. **changeNumberPresentedQuestion()** method is called by the study class to change the value *numPresentedQuestion* variable. The change can be random (from 1 to *maxPresentedQuestion*) by using the Random object (RandomGenerator) provided by Java API.

No	Variable's name	Type	Description
1	name	string	The name of the experiment
2	numQuestion	Integer	The number of questions will be asked on the experiment
3	numPresentedQuestion	Integer	The number of question presented to the participant on the experiment every phase of question-answer
4	questionOrder	String	If the value is RANDOM, then the question is picked randomly from a list of question, if the value is LINEAR then the question will be picked based on the order of the input file
5	randomPresentedQuestion	Boolean	If it the value is true, then on each phase the num of presented question will change randomly, explained more on changeNumberPresentedQuestion method
6	maxPresentedQuestion	Integer	This is the maximum number of presented question if the number of presented question is decided randomly
7	randomGenerator	Random	This is a random class that use to generate a random number, it is used inside changeNumberPresentedQuestion method

Table 4.1: variables inside the experiment class

4.1.3 Category Class

Category class is used to carry the questions objects. It has two main variables; *questions* and *askedQuestion*. *Questions* variable consist a list of questions. This variable is filled by the question from the input file. *AskedQuestion* variable consists of all the question that had been asked.

The class has two main methods *getRandQuestion()* and *getQuestion()*. These methods will be called during the quiz activity to put question object into activeQuestion variable. These methods are two different procedure to pick a question from *questions* variable. the former takes the question randomly while the latter takes the question based on the order of the input file. Java random class is used to generate random index (from 0 to the size of the *questions* variable). After the question is selected, it will get deleted from the *questions* variable and then it will be put it into *askedQuestion* variable.

4.1.4 Question Class

Table 4.2 shows the variables inside this class. The *question* class consist of the content of question; it's question text, answer link and answer. The class also contains the tracked variable (the variables are explained more on tracker section). The question can be two type MC or TEXT. MC means multiple choice, this question type will have multiple options on its answer choice, and the participant can choose one of them. while TEXT means that the participant needs to write the question. *representId* is generated when the questions are presented to the participant. this variable is unique and can be used to classify which questions are presented at the same time.

No	Variable's name	Type	Description
1	id	string	The id of the question.
2	text	string	The text of the question.
3	linkAnswer	string	the URL link to the answer page.
4	answer	string	(optional) the answer of the question.
5	participantAnswer	string	the answer of the participant during the quiz activity.
6	questionType	String	The type of the question. The value can be ”MC” or ”TEXT”.
7	representId	string	the random id generated when the question is presented during the quiz.
7	options	list of String	The answer option if the question is multiple choice.

Table 4.2: Variable inside Question class (without the tracked variable)

4.1.5 BoxNotification class

BoxNotification is used as a class name because the name Notification is already defined inside the Android framework. Most of the variable in this class is similar to the defined variable in the design section. Similar to the question class, the notification also contains *presentedID* which show on which question the notification is shown. The content of the notification can be set into *title* and *msgText* variable. The Notification can open an android application of twitter, facebook, instagram and web browser. The application should be installed on the phone, otherwise, the notification will open the url of the application on the web browser. To define which user will be shown if the user clicked the instagram, twitter or facebook notification. The user is defined by the user_id code which can be found on the profile of their social media. the url can also contain the URL link, then the application will open the web page. This app needs to be specified on the *app* variable and the *url* variable. The *url* variable needs to be filled with the user id of the twitter or instagram. Or it can be filled with http/https url to open web page. the class contains a show() 0method that will pop up the notification on the android phone. The mechanism and flow of the notification will be explained on next chapter.

4.2 Application flow

In this section, the flow of the application from the technical point of view is provided. The flow of the application can be seen on figure 4.2. The figure shows the flow of the application after input file is uploaded and the experiment just has started. As it seen in figure, the flow is divided into four scopes;

- **Setting** : the researcher can set some properties of the experiment or choose to start the experiment.
- **Experiment** : The participant conducts the quiz experiment.
- **Notification** : The notification can be shown during the quiz.
- **PostQuestion** : The participant presented with post questions.

To have a further understanding about the application flow, the activities and method on the flow chart are explained further in the following subsection.

4.2.1 Android Activity

The android application built upon multiple class activities. Each activity has a user Interface (UI) template. this template is saved into an xml file. The template consists of UI element, for example, button, text, etc. Its corresponding class activity will decide what will appear on the phone or what happens if a button is clicked. For example, answerActivity class will have activity_answer.xml template. The activity class is used to catch the event such us clicked or move to another application. This event is linked to an *event listener* methods.

The android application also contains a lifecycle which is default methods and activities that will be called every time. Figure 4.3 shows the android activity lifecycle (and, 2017). There are three methods inside the lifecycle that is used in this experiment application; **OnCreate()** is the first method to get called every time the activity start. **OnPause()** will be called if the phone move to another application. Lastly, **OnResume()** is called if the user open the application again after leaving the application.

4.2.2 expInitialActivity

The UI layout of this activity can be seen in figure B.1. The participant can click a button to begin the experiment or to set the properties of the experiment. On the **OnCreate()** method the **InputReader.read()** method is called firstly. This method will read the JSON input and compiled it into *Study* object. String input is compiled into JSON object by using *GSON* library. *GSON* is a serialization / deserialization library that is used to convert a string into a JSON object or another way around. (gso, 2017)

Then the JSON object compiled into The *Study* object. This object controls the quiz experiment and hold all the data, including the track variables. This object will be sent through the activities. *Intent* class of java is used to encapsulate the object and send to another activity. Because the *Study* object need to be encapsulated inside the Intent, java programming language requires every class sent by using an intent to implement *Serializable*.

4.2.3 expSetPropActivity

The UI layout of this activity can be seen in figure B.2. This activity is used to set some parameter of the experiment. In this activity, the researcher can pick which experiment to conduct, the name of the experiment, the id of the researcher and the id of the participant. The purpose of this activity is to make it easier for the researcher to conduct multiple experiments and multiple participants without uploading the input file again.

4.2.4 IntroActivity

The UI layout of this activity can be seen in figure B.3. This activity is used to show the information about the experiment to the participant before starting the experiment. This information is obtained from the preText and postText variables. the value of this variable will be converted into a html and shown to the participant.

4.2.5 ChooseCategoryActivity

The UI layout of this activity can be seen in figure B.4. In this activity the participant chooses which category he/she want to answer. The selected category name will then save in the *selectedCategoryName* variable inside study class. *selectedCategoryName* is used to in the experiment initialization to initialize *activeCatg* variable

4.2.6 QuestionActivity

The UI layout of this activity can be seen in figure B.5. In this activity, the question inside *activeQuestion* variables is shown to the participant. This activity will be called multiple time during the quiz experiment. The main function of this activity is to call **Study.runExperiment()** method. **Study.runExperiment()** is used to start or continue the quiz experiment. this method is explained in the section below.

4.2.6.1 Study.RunExperiment()

This method will be called every time the Quiz activity started. The main function of this methods are :

- Initialize the active experiment (*activeExp* variable) and active category (*activeCatg*) variable.
- Change the number of *presentedQuestion* (the variable is explained in the Experiment class section).
- Set the active questions (*activeQuest*) from the questions in the category.

Figure 4.4 shows the work flow of the method. The First step is to initialize the experiment by checking the active experiment (*activeExp*) variable. if it's empty then it's the beginning of the experiment, hence some variables needs be initialized. If it's mean that the experiment had been started (on-going) so new question is presented to the participant.

On the initialization, active experiment (*activeExp*) and active category (*activeCatg*) are initialized by calling **initializeExperiment()** method. The **initializeExperiment()** method will fetch the selected *experiment* and the *category* object from the study class variables based on the *selectedExperimentName* and *selectedCategoryName*.

Subsequently, the **setRandomPresentedQuestion()** is called, this method will set the value of *numPresentedQuestion*. If the researcher set randomPresentedQuestion to true in the input file, then **Experiment.changeNumberPresentedQuestion()** is called, this method will set numPresentedQuestion randomly On the other hand if it's false, then the numPresentedQuestion will be constant.

Next, **isExperimentIsStillGoing()** is called. This method make sure if the experiment is still on progress by checking the size of the *question* variable inside the *activeCatg*. If the size is still larger or similar then *numPresentedQuestion* then the experiment can be continued.

Lastly, the active question (*activeQuest*) is picked from the questions variable by calling **setActiveQuestion()**. this method filled *activeQuest* variable by fetching *question* object from the *question* variable inside *activeCatg* object. The question object will be picked randomly if the researcher set *questionOrder* to random. Otherwise, it will be picked linearly based on the input file order.

4.2.6.2 AnswerActivity

The UI layout of this activity can be seen in figure B.6. In this activity, the answer links are shown as a textview inside the UI layout. If the participant clicks the answer link then the **clickListener()** method inside the textview will open the answer page. A java class called *webview* is used as a browser. The *webview* will open the web page based on the URL of the answer link (*question.url*).

On the layout, two radio buttons are presented to the participant. These are the option whether the participant wants to return to see the question again or to continue to answer the question. If the participant chose to look at the question again then a special string is capsulated inside the *intent* object. This string is sent to question activity then send back to answer activity. It is also used to indicate if the participant looks at the question again. If the string is sent to answerActivity then the radio button "to see the question again" is not visible.

4.2.7 fillAnswerActivity

The UI layout of this activity can be seen in figure B.7. In this activity, the participant should answer the question by writing the answer on the editText UI. If the participant clicks next button then **saveAnswer()** method is called. this method will get the value of the editText and stored it on the *participantAnswer* variable inside the *question* object.

4.3 Notification mechanism

Figure 4.5 shows the mechanism of notification. As seen on the flowchart, the **checkNotification()** method is called inside the OnCreate event on the QuestionActivity, AnswerActivity and FillAnswerActivity. The method checks on every notification inside the *notifs* variable if the there is a notification that should be shown up based on *phase* and *shift* variable of the notification.

If there is a notification that needs to be shown up then the notification object is added into the *activeNotif* and it is deleted from the *notifs* variable of the study object.

The notification needs to wait for some millisecond before it can be shown up. The waiting time is defined in *timeToshow* variable inside the BoxNotification class. While the notification process waits, the main activity should keep working, so another process needs to be spawned a part of the main process. To accomplish it, *TimerService* class is used. This class will be spawn as a new process and it will sleep for *timeToShow* millisecond. After that, the service class will call the **BroadcastReceiver()**. The **BroadcastReceiver()** method is defined on all of activities and it simply calls **show()** method of the notification. The **Notifaction.show()** method is used to show the notification to the front end of the android screen. This method use *NotificationCompact.Builder* to build the notification layout. an *Intent* object is inserted inside the Builder object that contains what application to open. If the participant clicks the notification then the experiment application will be minimized. This event will call **OnPause()** method on the current activity and the android phone will open the intended application. The participant if the open the experiment application again, then the **OnResume()** method is called.

4.4 Tracker

The tracked variable are shown on the table 3.7. All of these variable are stored inside the *question* class as seen in the class diagram 4.1. Some of the variables are used to track the time in millisecond. To track the time the *stopWatch* class provided by java API is used. The *StopWatch* object is stored in the activity class because the *StopWatch* class is not serizable. . During the experiment the application is minimized (participant clicks the notification). Then event some Stopwatch object need to be paused. Inside **OnPause()** event listener the stopWatch is suspended, and it will resume again inside the **OnResume()** event listener.

As seen in the class diagram, each one of tracked variable has its own *stopWatch* object, for example, *stopWatchTTLQ* will track the time for TTLQ variable. To get the millisecond time of the stopwatch a **StopWatch.getTime()** method is called. Then to stored the tracked time, the **study.log()** method is called. This method pass two arguments; what variable to track and which stopwatch object is used to track it. for instance **study.log("TTLQ",stopWatchTTLQ)** will track the *TTLQ* variable and use *stopWatchTTLQ* to track the time. All of the tracked variables are then stored to the *question* object inside the *ActiveQuest* variable. How each variable is tracked is explained further in the subsection below. The explanation of the variables are explained on table 3.7.

4.4.0.1 TTLQ, lb_TTLQ and lookback

These variables is tracked inside the *questionActivity*. *StopWatchTTLQ* and *stopWatchTTLQ_lb* are used to track the time of this variable. The stopwatch object starts to count the time when **OnCreate()** method is called. The **Study.log()** method will be called to track the variable when the next button is clicked then the stopwatch is stopped. The lookback variable will have true value if the participant chose to see the question again.

4.4.1 TTLB and lb_TTLB

These variable are tracked inside the *AnswerActivity*. Similar with TTLQ, *stopwatchTTLB* and *stopWatchTTLB_lb* are used to track these variable. The stopwatch will start on **Oncreate()** method and it will be tracked when the participant clicks the next button.

4.4.2 visitedLinks, timeVisitedLink

These variable is tracked inside the *AnswerActivity*. *StopWatchLink* then will be started, and it are used to track the time participant have spent on each web page inside the *webview*. The mechanism of the tracking is shown in figure 4.6. Firstly, the prevUrl variable is initialized, this variable stored the previous link the *webview* had opened. This *webview* class has an event listener called **onPageFinished()** which will be called every time the web page has been finished loaded, for example when the participant clicks a new link and open a new web page. Every time **onPageFinished()** is called and when the participant clicks the next button then *updateVisitedLinks()* method is called. **updateVisitedLinks()** method then store the value of *prevUrl* to visitedLinks variable and how long the participant spent on the web page to *timeVisitedLinks*.

4.4.3 TTLA and TTLFA

These variable are tracked on the **fillAnswerActivty**. *TTLA* is tracked using *StopWatchTTLA*, while *TTLFA* is tracked differently because if there is more than one question then there will be multiple *editText* for the answer field, and each one of them needs to be tracked.

stopWatchTTLFA is made as a hashmap where the key is the id of the *editText* element. The id is made from the index of the *question* inside the *activeQuest* variable. The value of the *stopWatchTTLFA* is the stopwatch object correspond of each question and *editText*. On each *editText* element, the event listener called **OnFocusChangeListener()** is attached to it. This event listener will be called if there is a change of focus on the UI layout of the activity. For example, if the user clicks an editText then clicks another editText, then the event listener method gets the id of which editText was active before. The id then stored inside the *activeViewId* variable. By using *activeViewId* variable a stopwatch corresponding to the id will be picked from the hashmap.

4.4.4 numNotif, numNotifClicked and TTLN

As seen in figure 4.5. The QuestionActivity, AnswerActivity and fillAnswerActivity call **checkNotifaction()** method which will find the notification that should be shown up to the screen. If the notification is found than the activity call the **inceraseNumNotif()** method. This method will increase the *numNotif* variable by 1.

If the notificiation is clicked then the **inceraseNumNotifisClicked()** method will be called inside the **broadCastReceiver** method on the current activity class. This method will increase the number of *numNotifClicked* variable by 1.

NotifStopWatch object is used to track the TTLN time. After the participant clicks the notification than the application will be minimized and other application will be opened. the **OnPause()** event handler will be called just before the application is minimized. then inside the event handle a **Study.startLogNotif()** method is called. The **Study.startLogNotif()** method will start the notifStopWatch object. Then after the participant return to the experiment application the **onResume()** method is called. This method will call **Study.stopLogNotif()** method. The **Study.stopLogNotif()** method then track the TTLN variable inside the active notification by using the *notifStopWatch* object.

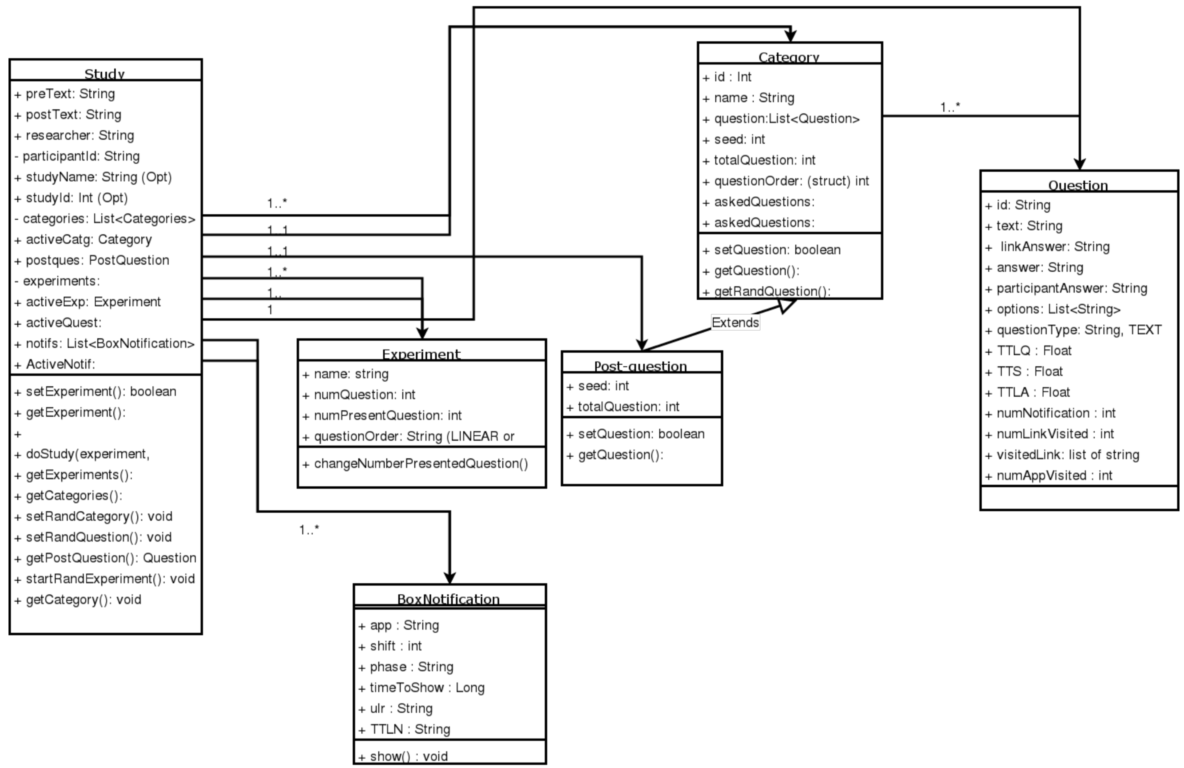

Figure 4.1: Class diagram of the application

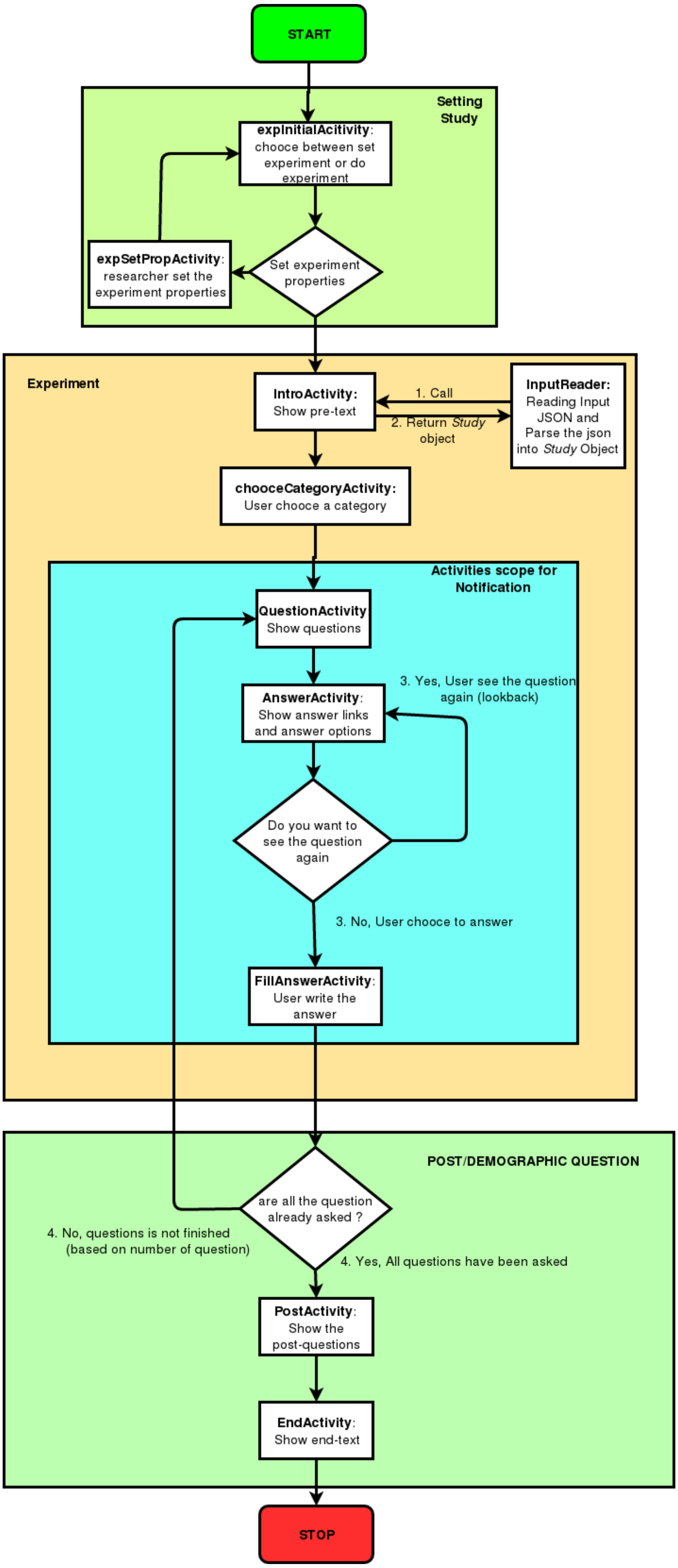

Figure 4.2: The flow of the application

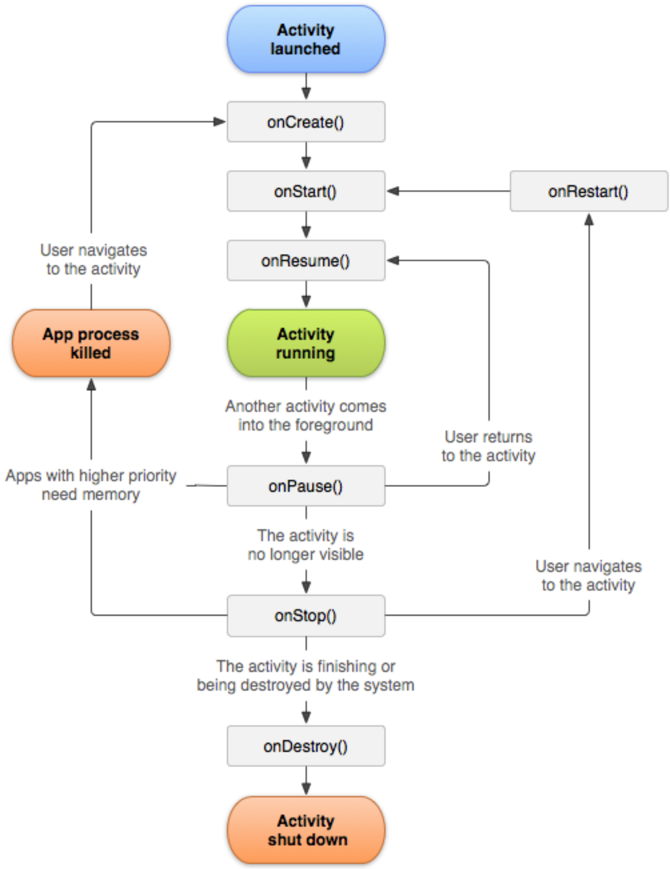

Figure 4.3: The lifecycle of android application

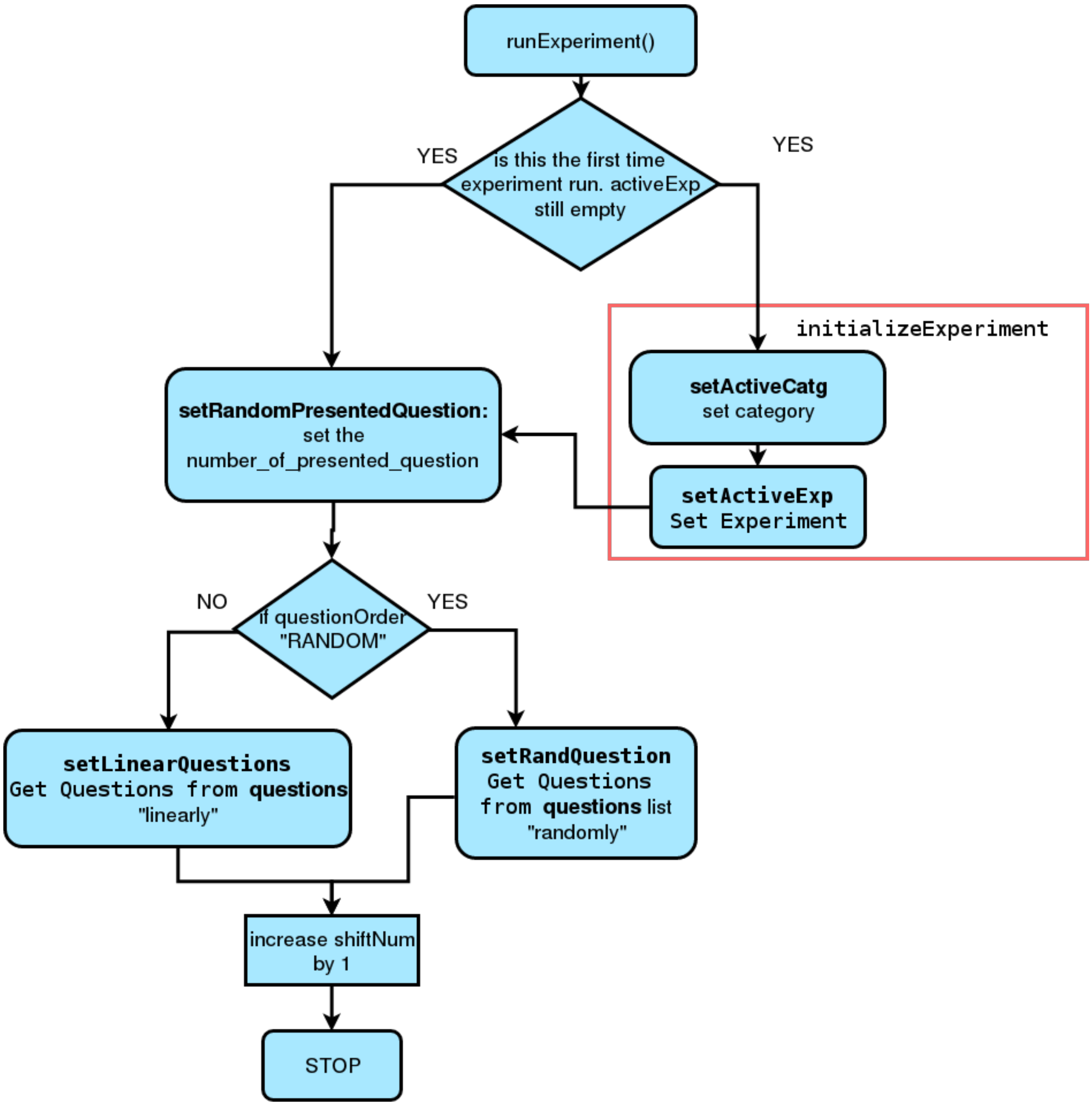

Figure 4.4: Flow chart of runExperiment method

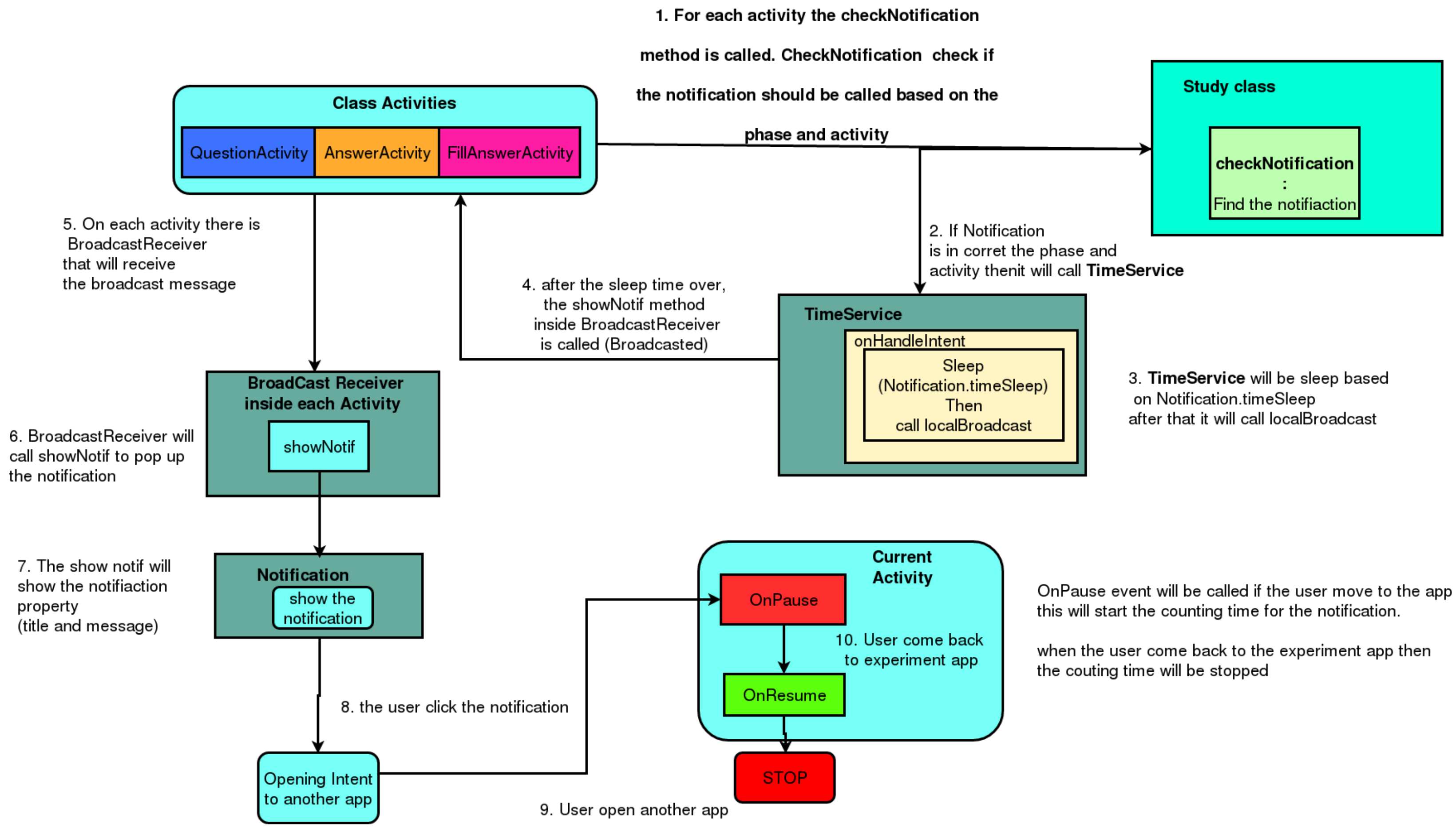

Figure 4.5: The flow of the notification

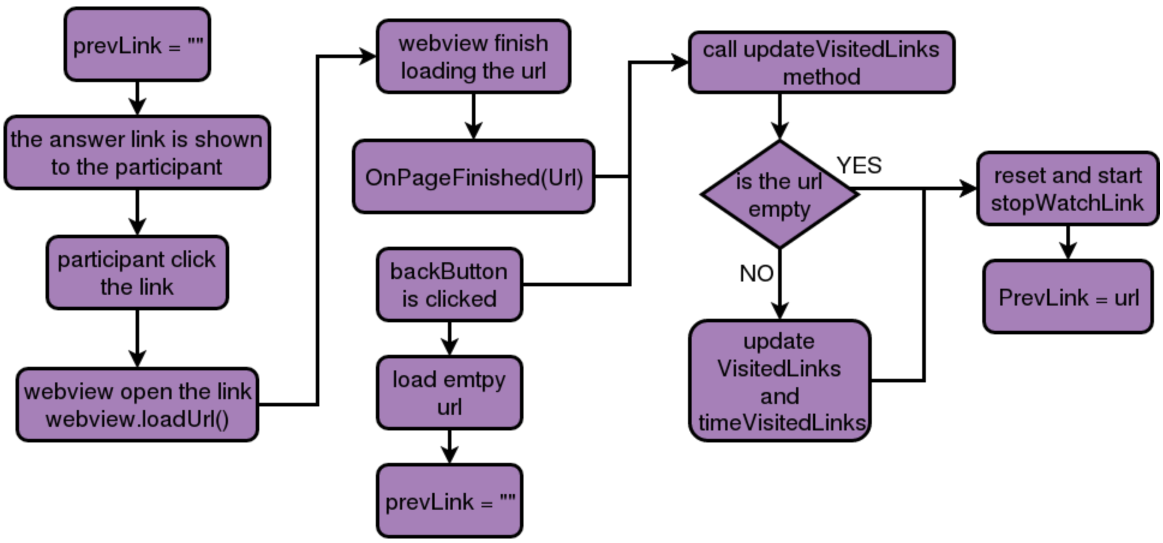

Figure 4.6: tracker mechanism for webview

Chapter 5

Experiment Result and Discussion

> Human memory is short and terribly fickle.
>
> *Janine di Giovanni*

This chapter provides the result and the analysis of the experiment. Three hypothesis are analyzed;

- Do participant experience the failure of prospective memory while using the smartphone?
- Is failure of prospective memory is more likely to happen with two intentions rather than one; intentional load matters
- Do increasing number of notification influence prospective memory error?
- Does mentally moving through event boundary increase the likeliness to experience failure of prospective memory?

5.1 Prospective memory error on smartphone

5.1.1 Experiment result

In this experiment, the data from the participants from the three studies were combined. Table 5.1 shows who many participants believe that they had experienced prospective memory error, and how many participants actually experience the prospective memory error during the experiment. The participant was asked if they believe that they experienced prospective memory error using the first question on Table 3.2. The actual experience of memory error was calculated by looking if the people forget the questions during the experiment.

	Experiment 1 (n=4)	Experiment 2 (n=11)	Experiment 3 (n=3)
How many participant believed they have experience prospective memory error	0	5	2
How many participant actually experienced prospective memory error	3	8	2

Table 5.1: Number of participant from all the studies who believed they have experince prospective memory error and the actual result of the experiment

5.1.2 Discussion

Most of the participant did not believe that they have experienced the prospective memory error. In contrast, the output shows that almost 70% of the participant actually experienced the prospective memory error. We can argue that the participant made an intention for looking the answer before clicking the answer link, but after reading the answer page they lost their original intention. As a result, they forget the content of the question, and they experience prospective memory error. The result shows that while using a smartphone a person has a high probability of experiencing prospective memory error. This experiment supports the result of Carlson's experiment.

5.2 The effect of multiple intention

5.2.1 Experiment result

This section shows the result from the second study. On the second study, one or two questions were presented to the participant randomly. Table 5.2 shows the total number of times the participant forgot the question and decided to see it again (lookback). Each row shows how many times they did a loopback, percentage of loopback (p) and the total of the question. The number of presented question was random, so the number of times the participant presented with one or two question are not similar. Thus the percentage (p) is calculated by dividing the number of loopback with the frequency of one or two question presented (N). The table shows nine people forgot the question, six of them forgot more frequently if two questions are presented. The last row shows that they forgot 31% of the time if presented with two questions. While they forgot only 8% of the time if they presented with only one question.

Figure 5.1 shows how long in millisecond each participant spent to write the answer (TTLFA). The horizontal axis is the 11 participants and the vertical axis shows the duration of writing. It shows that 63% (7 out of 11) spent longer time writing the answer if two questions were presented each time.

Figure 5.2 shows the general mean of the writing time between one or two questions from all the participant. It shows that if two questions were presented, then it will take longer time for the participant to write the answer.

Figure 5.3 shows the mean each participant spent on finding the answer on the answer page. The top plot was calculated on the first time they see the answer page. While the lower plot was calculated the second time they see the answer page after they decided to see the question again (loopback). Surprisingly the top plot shows that almost half of the participants (4 out of 11) spent significantly longer time for one questions. However. The lower plot shows that most of the participant spent longer time for two question.

5.2.2 Discussion

In this experiment, the intention of the participant was to find the answer and to write it thus the number of questions presented is the number of intention need to be retained by the participant. If the participant forget the question, we can argue that they experience loss of intention, as a result, they experience prospective memory error (Reason, 1985).

Using this result we are trying to see if an increasing the number of intention will make people more likely to experience failure of prospective memory (is the intentional loads matter ?). Based on the table 5.2, a person more probable to experience prospective memory error if the amount of the intention was higher. The result of this study shows that the amount of intentional loads are important component of prospective memory.

Furthermore, the result in figure 5.1 and figure 5.2 shows that the increasing amount intentional loads also made the person harder to recall the content of the intention. the recall time is presented as the time participant write the answer. On this analysis, the intention was to answer the question and it was formed after the participant found the answer.

In addition, based on figure 5.3, the increasing amount of intention also increase the time the participant spent on looking for the answer. This result shows that the number of intention decrease their cognitive performance. This was probably because the increase of intentions reduced the level of attention while doing the task, and made

Participant	Look the question again (Lookback)	
	one question	two question
1	0 (p=0%, N=2)	0 (p=0%, N=8)
2	1(p=16%, N=6)	4 (p=100%, N=4)
3	0 (p=0%, N=4)	2 (p=33%, N=6)
4	1 (p=25%, N=4)	0 (p=0%, N=6)
5	0 (p=0%,N=0)	0 (p=0%, N=10)
6	0 (p=0%, N=6)	4 (p=100%, N=4)
7	0 (p=0%, N=4)	2 (p=33%, N=6)
8	1 (p=25%, N=4)	4 (p=66%, N=6)
9	0 (p=0%, N=4)	4 (p=66%, N=6)
10	1 (p=10%, N=10)	0 (p=0%, N=0)
11	0 (p=0%, N=2)	0 (p=0%, N=8)
Total	4 (p=8%,n=46)	20 (p=31%, n=64)

Table 5.2: The number of lookback and its percentage (p) between one or two question from study 2

	Mean time to write the answer (TTLFA)	Mean time spent on looking for an answer	Forget the question (lookback) percentage
No Notifications	6.267 second	59.875 second	17%
One Notification	7.292 second	69.517 second	11%
Two Notifications	7.304 second	76.699 second	21%

Table 5.3: The result of different number of notification received by the participant from all studies

the participant spent longer time to find the answer.

5.3 The effect of notification

5.3.1 Experiment Result

Table 5.3 shows the mean time participant spent on writing the answer (TTLFA), mean time finding the answer and the percentage of looking the question again (lookback). The variables are grouped by the number of notification received. It shows that increase in the number of notification make people spent more time writing and finding the answer. However, the number of notification does not have correlation with the percentage of lookback.

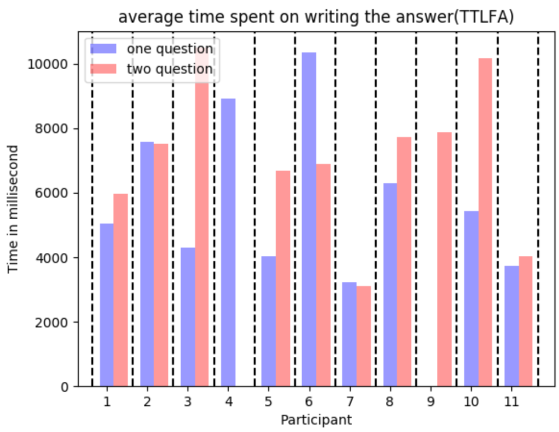

Figure 5.1: Mean time filling the answer of each participant between one or two question in study 2

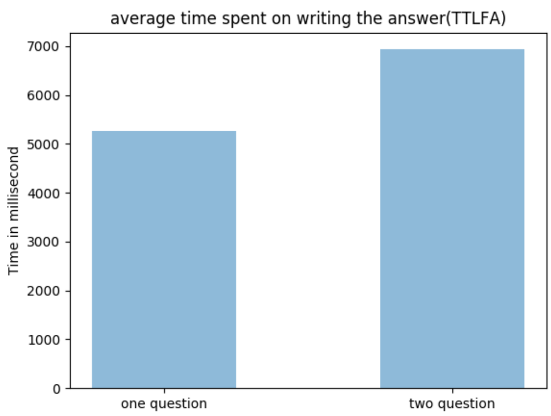

Figure 5.2: Mean time filling the answer of all the participants between one or two question in study 2

5.3.2 Disscussion

This result shows that the notification does not have any effect on the prospective memory error. Table 5.3 shows that the increasing number of notification made the participant spent longer time to write and to find an answer. Because in this experiment the notification appeared when the participant looks at the question and find the answer. Then the

If the notification appeared when the person is looking for the answer then the result

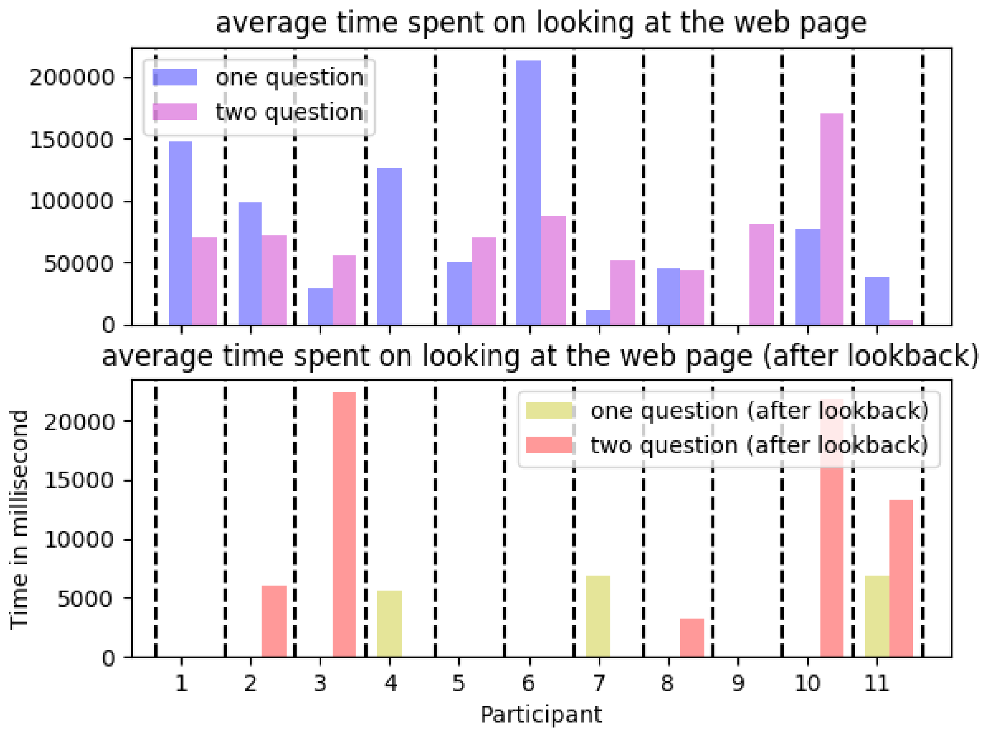

Figure 5.3: Mean time in spent finding for an answer between one or two questions

is consistent with the time needed to pay attention and to dismiss the notification. Thus we can not draw a conclusion based on the time spent looking for the answer.

However, the notification did not appear when a participant was writing the answer hence we can analyze this variable. In this analysis, we argue that the intention is writing the answer and the content is the answer itself. Therefore the notification makes them harder to recall the content of the intention. Probably because the notification act as a distraction that lower the level of attention when the intention is framed to the memory.

5.3.3 Event boundary on prospective memory

5.3.4 Experiment Result

The bar chart 5.4 shows the mean time (in millisecond) the participants spent finding the answer on the answer page. The chart shows the participant who forgets the question (lookback) had spent more time finding the answer before decides to see the question again. The bar chart 5.5 and 5.6 shows how many time the participant saw question again (lookback) on each question, it shows the data from first and the third study respectively. the bar charts show that the more participants forget the question on the third study. Table 5.4 shows the number of web page participant visit by click link inside the web page, and the number of times participant forget the question (lookback). for the visited link that has zero value, this is probably because the participant did not click the answer links or the application fails to track the data.

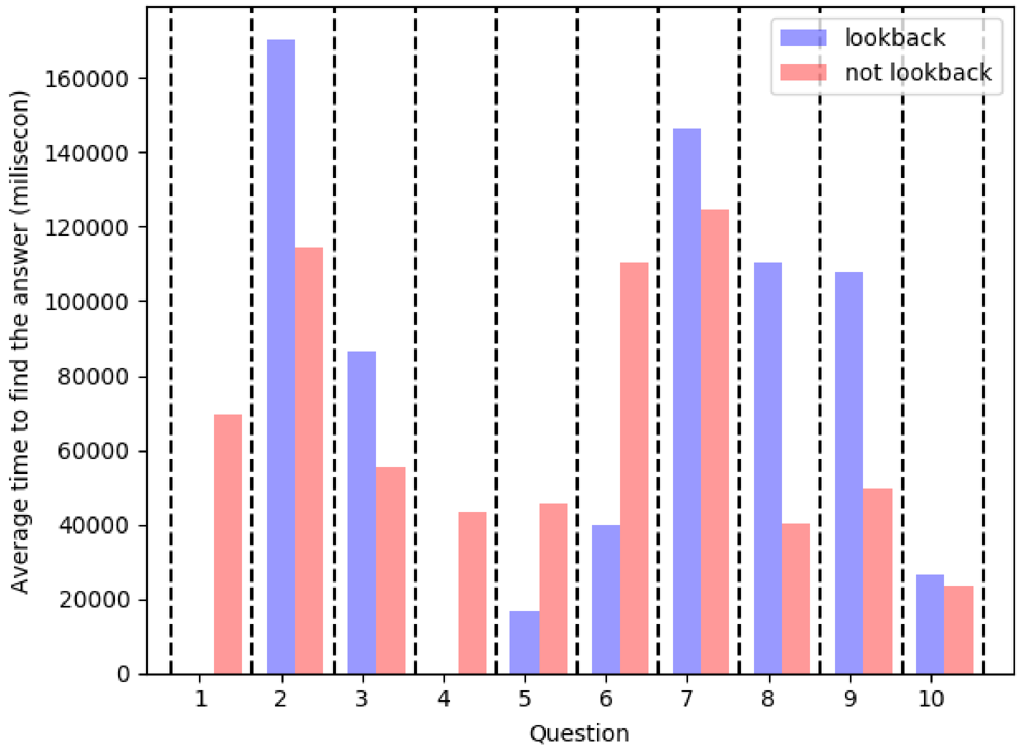

Figure 5.4: Mean time in spent looking for an answer between lookback and non-lookback in all studies

5.3.5 Discussion

A participant experienced a transition through event boundary after they move from one view to another view on the smartphone. In this discussion the transition happened two times; when the participant saw the question then clicked the button then the application changed to answer link view, and when the person clicked the answer link then the application changed to the answer page view. During this transition, the participant still retained the intention which was looking for the answer.

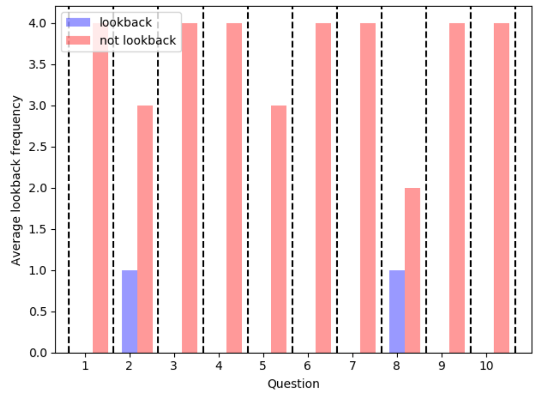

Figure 5.5: Frequency of lookback of the participant on study 1

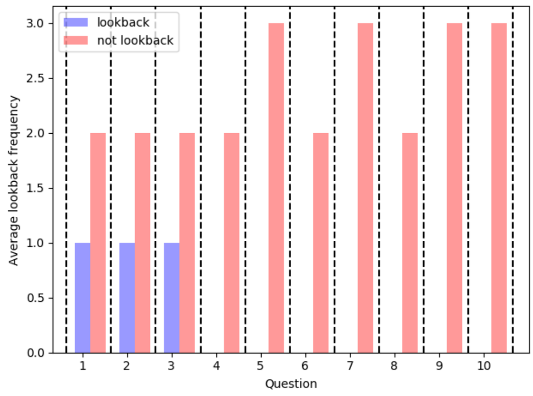

Figure 5.6: Frequency of lookback of the participant on study 3

The result of the bar chart 5.4 shows that the participants who experienced prospective memory error had spent more time finding the answer. We can argue that when they were reading the answer page, they forgot the intention (lost intention) or forgot the content of the intention (detached intention) which result in prospective memory error. This happened probably because the amount of new information they received during reading reduce the retention level of the intention (Reason, 1985). We can also argue that a new information mentally create a new event boundary. So the longer time they spent reading the web page, more mental transition happened. This moving through more event boundary makes a person more likely experience prospective memory error.

However, if we consider moving through a new web page as a transition between event boundary. Table 5.4 shows that there was no correlation between a mental tran-

Number of visited links	number of times participant forget the question (lookback)(N=192)	number of times participant,remember question(N=192)	SUM
0	9	6	15
1	35	135	170
2	1	5	6
3	0	1	1
SUM	45	147	192

Table 5.4: Number of visited page and how many times participant forget or remember the question from all the studies

sition to prospective memory error. But this view is very weak since most of the participant only visited one web page.

To understand the effect of physical transition through event boundary, we try to investigate the frequency of prospective memory error between first in the third study. Bar chart 5.5 and 5.6 shows that the participant on the third study forgot the question more frequently than the first study. However, we can only see the probability that physical transition will increase the likeliness of prospective memory failure, but we cannot draw a strong correlation because the sample is very small. This possibility contradicts the result from Carlson experiment. Over all, the results on this discussion section support the event horizon model proposed by Radvansky and Copeland (2006).

Chapter 6

Conclusion and Suggestion

6.1 Conclusion

Edinburgh is a city full of sweet memories.

We have successfully built an application that can be used to conduct prospective memory error experiment. The application support configurable setting which allows an experimenter to change experiment properties and to conduct an experiment using a large sample. The application has been made publicly available on the github repository.

By using the application, three studies have been successfully conducted. This study is based on conduct Carlson's studies. From the result of the studies, we can draw a three conclusion. Firstly, the participant experiences prospective memory error while using the smartphone. Secondly, the increasing number of intention make people more likely to experience prospective memory error (intentional loads matter). Lastly, Moving through event boundary mentally or physically *probably* increase the likeliness of prospective memory failure. These first two results support the result of Carlson's experiment. While the last one contradict their result.

6.2 Future suggestion

6.2.1 Experiment application suggestion

To make a more dynamic and ready to public use, the experiment application still have a lot of features that need to be implemented. The application should have a more user-friendly setting so the experimenter can easily set the experiment properties and upload the input file. On the analysis of the data, it's quite hard to analyze the questions and answer object because there is no field that shows the order of the question, so the field that shows the order of the question should be made.

To have better understanding about the event boundary, the question and the answer page presented should be more complex and require the participant to search the answer by clicking multiple links on the answer page. Moreover, The application should able to track the movement of the participant so the further analysis can be made on the effect of physical movement on the failure of prospective memory.

6.2.2 Experiment design suggestion

On the future, I hope that the experiment can be conducted on the bigger sample of participants, and the smartphone of the experiment can be the personal smartphone of the participant. I Also think rather than using a smartphone, the experiment can be conducted by using virtual reality (VR) so the immersion that the participant experience will be much higher, and the study can be more precise on investigating the prospective memory error in everyday life.

6.2.3 Futher investigation

I also suggest that there is a further investigation on the intention and how it stored on the memory. Also, there should be an investigation whether an intention that is correlated with each other, for example buying a jeans and shirt, is easier to remember than buying a cake and a shirt. The motivation to formed the intention should also be investigated, such as giving the participant reward if they do a better prospective memory task.

Appendix A

List of documents used in the experiment

Consent Form

Study title: "What was I meant to do again"(Exploration of event boundary on failure of prospective memory)

June, 2017

	Please tick box
1. I confirm that I have read and understood the Participant Information Sheet [date: 22 June 2017] for the above study and there is no reason I should not take part. I have had the opportunity to consider the information and ask questions, and have had these answered satisfactorily.	☐
2. I understand that my participation is entirely voluntary and I am free to withdraw at any time without giving a reason.	☐
3. I agree to take part in this study.	☐

Name of volunteer	Date	Signature
Name of researcher	Date	Signature

School of Informatics, 11 Crichton Street, Midlothian , Edinburgh, EH8 9LE

1

Figure A.1: Consent form used in the experiment

Participant Information Sheet

Study title: "What was I meant to do again"(Exploration of event boundary on failure of prospective memory)

August 17, 2017

We are inviting you to take part in a research study This study is part of Aldy Syahdeini's MSc Research at the University of Edinburgh. The supervisors are Dr Maria Wolters and Ms Catherine Crompton.

Before you decide if you want to participate, we would like to explain to you why we are doing the study and what this will involve. Please read this document carefully. If you have any questions, please ask us, and we will be happy to explain.

1 Why is this research being undertaken? What am I asked to do?

The purpose of this research is to look at what happens when you need to go to a new website in order to look up some information. We will ask you to look up the answers to several questions, and log the web sites you visit. You may also get messages and reminders while you go through the questions. There will be eight questions in total. One or two question will be presented at a time. Each question will be followed by an information page that contains the answer. Look up the answer, and then go back to the application and type in the answer on the next page.

2 What will happen to my data?

Only members of the research team will have access to your data. Your data will be anonymized and used mainly in intergroup comparisons. None of your personal information will be identifiable in any publications resulting from the study.

3 Do I have to take part? What if I change my mind?

Participation in this study is entirely voluntary. You can refuse to take part or withdraw from the study at any time without having to give a reason. Such a decision has no adverse implications for you. If you have any questions or require further information please do not hesitate to contact:

1

Dr Maria Wolters
Email: maria.wolters@ed.ac.uk
+44 (0) 131 650 2732

Mr Aldy Syahdeini
s1575408@sms.ed.ac.uk
+44 (0) 754 835 3098

Catherine Crompton
s0675382@sms.ed.ac.uk

University of Edinburgh
School of Informatics
11 Crichton Street
Edinburgh, Midlothian, EH8 9LE

Figure A.2: Information sheet used in the experiment

Appendix B

Figures of UI layout

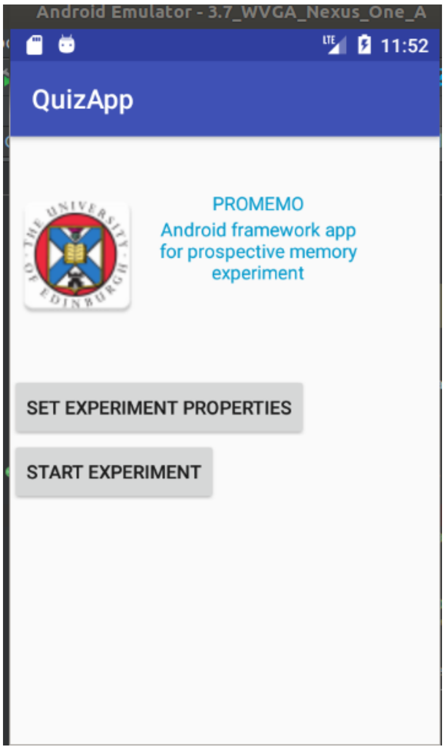

Figure B.1: expInitialActivity UI layout

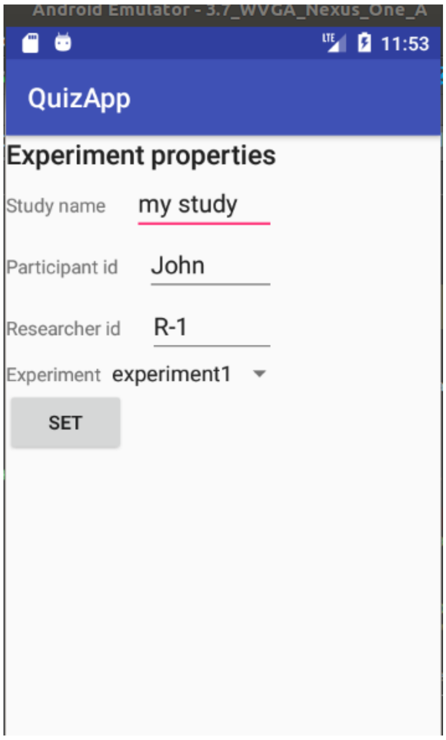

Figure B.2: expSetPropActivity UI layout

Figure B.3: IntroActivity UI layout

Figure B.4: ChooseCategoryActivity UI layout

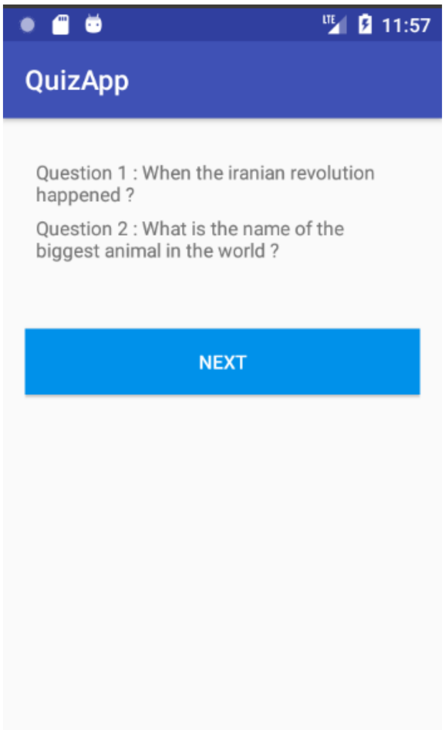

Figure B.5: QuestionActivity UI layout

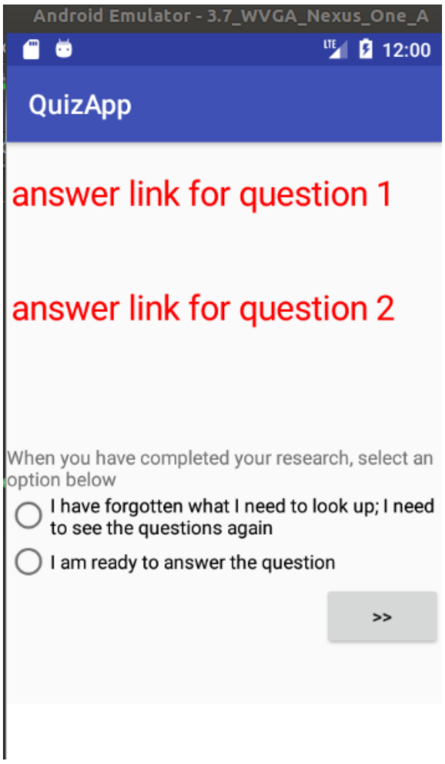

Figure B.6: AnswerActivity UI layout

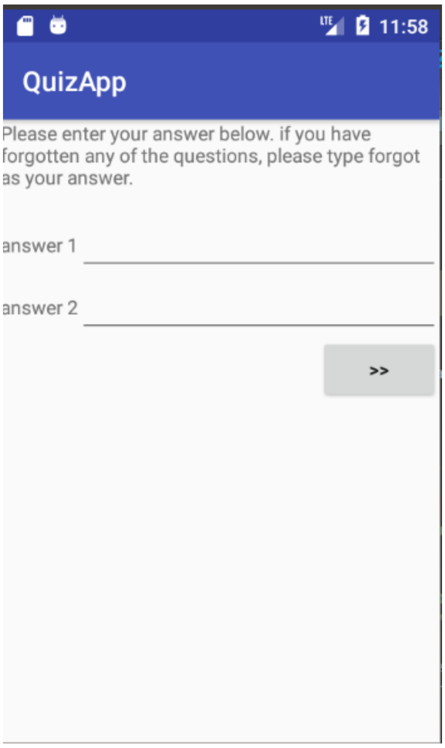

Figure B.7: fillAnswerActivity UI layout

Bibliography

(2017). Android activity lifecycle. `https://developer.android.com/guide/components/activities/activity-lifecycle.html`. Accessed: 2017-08-01.

(2017). Gson library. `https://github.com/google/gson`. Accessed: 2017-08-01.

(2017). Json description. `http://www.json.org/`. Accessed: 2017-08-01.

Baddeley, A. and Wilkins, A. (1983). *Taking memory out of the laboratory.*, pages 1–17. Academic Press.

Brandimonte, M., Einstein, G., and McDaniel, M. (1996). *Prospective Memory: Theory and Applications*. L. Erlbaum.

Cockburn, J. (1995). Task interruption in prospective memory: A frontal lobe function? *Cortex*, 31(1):87–97.

Cockburn, J. and Smith, P. T. (1994). Anxiety and errors of prospective memory among elderly people. *British Journal of Psychology*, 85(2):273–282.

Cohen, G. and Conway, M. (2008). *Memory in the Real World*. Psychology Press.

Crovitz, H. F. and Daniel, W. F. (1984). Measurements ofevery day memory:toward the prevention of forgetting. *Bulletin of the Psychonomic Society*, 5:413–414.

Curiel, J. M. and Radvansky, G. A. (2002). Mental maps in memory retrieval and comprehension. *Memory*, 10(2):113–126.

Einstein, G., J. Holland, L., Mcdaniel, M., and Guynn, M. (1992). Age-related deficits in prospective memory: The influence of task complexity. 7:471–8.

Einstein, G. O. and Mcdaniel, M. A. (1990). Normal aging and prospective memory. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, 16(4):717–726.

Einstein, G. O. and McDaniel, M. A. (2005). Prospective memory. *Current Directions in Psychological Science*, 14(6):286–290.

Ellis, J. (1996). Prospective memory or the realization of delayed intentions: A conceptual framework for research. In Brandimonte, M., Einstein, G., and McDaniel, M., editors, *Prospective Memory: Theory and Applications*. L. Erlbaum.

Gauld, A. and Shotter, J. (1977). *Human Action and Its Psychological Investigation*. Routledge & Kegan Paul, Limited.

Groot, Y. C., WILSON, B. A., EVANS, J., and WATSON, P. (2002). Prospective memory functioning in people with and without brain injury. *Journal of the International Neuropsychological Society*, 8(05).

Guez, J. and Naveh-Benjamin, M. (2013). The asymmetrical effects of divided attention on encoding and retrieval processes: A different view based on an interference with the episodic register. *PLOS ONE*, 8(9):1–14.

Guynn, M. J., McDaniel, M. A., and Einstein, G. O. (1998). Prospective memory: When reminders fail. *Memory & Cognition*, 26(2):287–98.

Haenggi, D., Kintsch, W., and Gernsbacher, M. A. (1995). Spatial situation models and text comprehension. *Discourse Processes*, 19(2):173–199.

Heckhausen, H. and Beckmann, J. (1990). Intentional action and action slips. *Psychological Review*, 97(1):36–48.

Heckhausen, H. and Kuhl, J. (1985). From wishes to action: The dead ends and short cuts on the long way to action. In Frese, M. and Sabini, J., editors, *Goal Directed Behavior: The Concept of Action in Psychology*, pages 10–134. L. Erlbaum Associates.

Jacoby, L. L., Woloshyn, V., and Kelley, C. (1989). Becoming famous without being recognized: Unconscious influences of memory produced by dividing attention. *Journal of Experimental Psychology: General*, 118(2):115–125.

Kliegel, M. and Martin, M. (2003). Prospective memory research: why is it relevant. *International Jonral of psychology*, 38(4):193–194.

Kurby, C. A. and Zacks, J. M. (2008). Segmentation in the perception and memory of events. *Trends in Cognitive Sciences*, 12(2):72–79.

Kvavilashvili, L. and Ellis, J. (1996). Varieties of intention: Some distinction and classifications. In Brandimonte, M., Einstein, G., and McDaniel, M., editors, *Prospective Memory: Theory and Applications*, pages 23–51. L. Erlbaum.

McDaniel, M. and Einstein, G. (2007). *Prospective Memory: An Overview and Synthesis of an Emerging Field*. Cognitive psychology program. SAGE Publications.

McDaniel, M. A., Robinson-Riegler, B., and Einstein, G. O. (1998). Prospective remembering: Perceptually driven or conceptually driven processes? *Memory & Cognition*, 26(1):121–134.

Mcgann, D., Ellis, J. A., and Milne, A. (2002). Conceptual and perceptual processes in prospective remembering: Differential influence of attentional resources. *Memory & Cognition*, 30(7):1021–1032.

Radvansky, G. A. (2012). Across the event horizon. *Current Directions in Psychological Science*, 21(4):269–272.

Radvansky, G. A. and Copeland, D. E. (2006). Walking through doorways causes forgetting: Situation models and experienced space. *Memory & Cognition*, 34(5):1150–1156.

Radvansky, G. A., Copeland, D. E., and Zwaan, R. A. (2003). Brief report: Aging and functional spatial relations in comprehension and memory. *Psychology and Aging*, 18(1):161–165.

Radvansky, G. A., Krawietz, S. A., and Tamplin, A. K. (2011). Walking through doorways causes forgetting: Further explorations. *The Quarterly Journal of Experimental Psychology*, 64(8):1632–1645.

Radvansky, G. A., Tamplin, A. K., and Krawietz, S. A. (2010). Walking through doorways causes forgetting: Environmental integration. *Psychonomic Bulletin & Review*, 17(6):900–904.

Reason, J. T. (1979). Actions not as planned: The price of automatization. In Underwood, G. and Stevens, R., editors, *Aspects of Consciousness*, pages 1–67. Academic Press.

Reason, J. T. (1985). Lapses of attention on everyday life. In Parasuraman, I. R. and Davies, D. R., editors, *Varieties of attention*, pages 10–134. New York Academic Press.

Scullin, M. K., Bugg, J. M., and McDaniel, M. A. (2012). Whoops, i did it again: Commission errors in prospective memory. *Psychology and Aging*, 27(1):46–53.

Searle, J. (1983). *Intentionality: An Essay in the Philosophy of Mind*. Cambridge paperback library. Cambridge University Press.

Shorrock, S. T. (2005). Errors of memory in air traffic control. *Safety Science*, 43(8):571–588.

Swallow, K. M., Zacks, J. M., and Abrams, R. A. (2009). Event boundaries in perception affect memory encoding and updating. *Journal of Experimental Psychology: General*, 138(2):236–257.

Wilkins, A. and Baddeley, A. (1978). *Remembering to recall in everyday life: an approach to absentmindedness.*, page 247. Academic Press.

Winograd, E. (1991). Memory in the real world. gillian cohen. lawrence erlbaum associates, hove and london, 1989. no. of pages: 247. ISBN 0-86377101-7 (paperback). price: $19.95. *Applied Cognitive Psychology*, 5(5):247.