

Jurani App

Software Requirements Specification

Version 1.0

31 Maret, 2016

Angellica Octavianzy Sabrina T.

Kenneth Mohamad Albany Hakim

Mahendra Sunt Servanda

Sandy Febriant Eka Putra

Revision History

Date	Description	Author	Comments
<date>	<Version 1>	<Your Name>	<First Revision>

Document Approval

The following Software Requirements Specification has been accepted and approved by the following:

Signature	Printed Name	Title	Date
	<Your Name>	Lead Software Eng.	
	A. David McKinnon	Instructor, CptS 322	

Table of Contents

REVISION HISTORY

DOCUMENT APPROVAL

1. INTRODUCTION

- 1.1 PURPOSE
- 1.2 SCOPE
- 1.3 DEFINITIONS, ACRONYMS, AND ABBREVIATIONS
- 1.4 REFERENCES
- 1.5 OVERVIEW

2. GENERAL DESCRIPTION

- 2.1 PRODUCT PERSPECTIVE
- 2.2 PRODUCT FUNCTIONS
- 2.3 USER CHARACTERISTICS
- 2.4 GENERAL CONSTRAINTS
- 2.5 ASSUMPTIONS AND DEPENDENCIES

3. SPECIFIC REQUIREMENTS

- 3.1 EXTERNAL INTERFACE REQUIREMENTS
 - 3.1.1 *User Interfaces*
 - 3.1.2 *Hardware Interfaces*
 - 3.1.3 *Software Interfaces*
 - 3.1.4 *Communications Interfaces*
- 3.2 FUNCTIONAL REQUIREMENTS
 - 3.2.1 *<Functional Requirement or Feature #1>*
 - 3.2.2 *<Functional Requirement or Feature #2>*
- 3.3 USE CASES
 - 3.3.1 *Use Case #1*
 - 3.3.2 *Use Case #2*
- 3.4 CLASSES / OBJECTS
 - 3.4.1 *<Class / Object #1>*
 - 3.4.2 *<Class / Object #2>*
- 3.5 NON-FUNCTIONAL REQUIREMENTS
 - 3.5.1 *Performance*
 - 3.5.2 *Reliability*
 - 3.5.3 *Availability*
 - 3.5.4 *Security*
 - 3.5.5 *Maintainability*
 - 3.5.6 *Portability*
- 3.6 INVERSE REQUIREMENTS
- 3.7 DESIGN CONSTRAINTS
- 3.8 LOGICAL DATABASE REQUIREMENTS
- 3.9 OTHER REQUIREMENTS

4. ANALYSIS MODELS

- 4.1 SEQUENCE DIAGRAMS
- 4.2 STATE-TRANSITION DIAGRAMS (STD)
- 4.3 DATA FLOW DIAGRAMS (DFD)

5. CHANGE MANAGEMENT PROCESS

A. APPENDICES

- A.1 APPENDIX 1
- A.2 APPENDIX 2

1. PENDAHULUAN

Di awal pembuatan *Software Requirement Specification (SRS)* tentunya dipaparkan terlebih dahulu bagaimana latar belakang dan apa tujuan dari pembuatan aplikasi. Tidak hanya itu, dalam pendahuluan ini juga dijelaskan ruang lingkup batas pembahasan *software*, daftar istilah yang digunakan selama proses dokumentasi, dan juga referensi-referensi yang dijadikan sebagai bahan rujukan. Adapun semua pembahasan tersebut ditujukan untuk mendukung proses penggambaran fungsionalitas dari aplikasi, pemaparan spesifikasi kebutuhan, analisis perancangan, serta implementasi program yang nantinya akan digunakan oleh khalayak umum.

The introduction to the Software Requirement Specification (SRS) document should provide an overview of the complete SRS document. While writing this document please remember that this document should contain all of the information needed by a software engineer to adequately design and implement the software product described by the requirements listed in this document. (Note: the following subsection annotates are largely taken from the IEEE Guide to SRS).

1.1 Purpose

Tujuan pembuatan dokumen ini adalah untuk memberikan gambaran secara rinci dari aplikasi Jurani. Dokumen ini akan menjelaskan mengenai tujuan dan fitur dari aplikasi, antarmuka aplikasi, apa saja yang dapat dilakukan oleh aplikasi tersebut, serta batasan-batasan yang ada di dalam aplikasi. Dokumen ini diperuntukkan bagi pemegang kepentingan dan pengembang aplikasi<<bla... bla.....>>

1.2 Scope

Jurani dibuat untuk mendukung kebahagiaan pada pengguna dan kerabatnya dengan memberikan pengalaman bermain *truth or dare* yang modern dengan aplikasi Jurani.

Berawal dari kegelisahan ketika momen kumpul bersama keluarga/kolega/shabat/rekan bisnis yang sering kehabisan topik atau para kerabat yang beralih dengan ponselnya, sehingga sering menghasilkan momen yang kurang berkesan. Maka dari itu kebutuhan akan kegiatan pendukung keakraban/*ice breaking activity* muncul, dan dair sini lah ide Jurani muncul.

Jurani adalah aplikasi yang mampu membuat momen berkumpul atau bertemu lebih akrab dengan melakukan permainan bersama. Pengguna dapat menggunakannya bersama sahabat, kolega, keluarga atau rekan bisnis.

Adapun fitur-fitur dari aplikasi Jurani ini adalah sebagai berikut:

1. Memasukkan jumlah dan nama pemain
2. Melakukan pengundian untuk menentukan siapa yang mendapat tantangan jujur atau berani di antara pemain yang ada
3. Melakukan pengundian terhadap tantangan yang terpilih baik itu tantangan jujur atau berani

1.3 Definitions, Acronyms, and Abbreviations

Istilah	Arti

1.4 References

IEEE. *IEEE Std 830-1998 IEEE Recommended Practice for Software Requirements Specifications*. IEEE Computer Society, 1998.

1.5 Overview

Bab selanjutnya, bagian *general description* dari dokumen ini menggambarkan fungsionalitas dari aplikasi. Bagian ini menjelaskan kebutuhan informal dan digunakan untuk membuat sebuah konteks untuk spesifikasi kebutuhan system pada bab selanjutnya.

Bab ketiga, bagian *requirement spesification* dari dokumen ini diutamakan untuk pengembang dan mendeskripsikan hal teknis dari detail fungsionalitas produk.

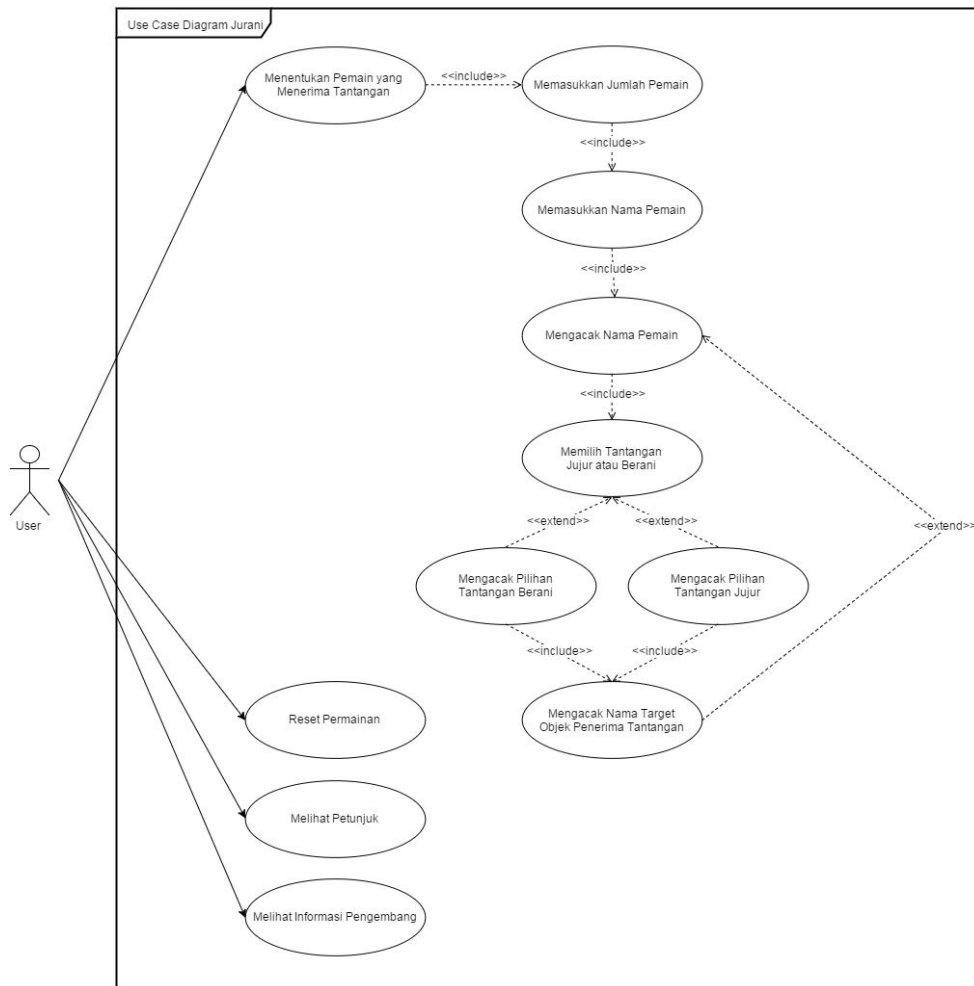
.....

2. General Description

This section of the SRS should describe the general factors that affect 'the product and its requirements. It should be made clear that this section does not state specific requirements; it only makes those requirements easier to understand.

2.1 Functional Requirements Spesification

Bagian ini menggambarkan *use case* dari aplikasi Jurani. Adapun yang menjadi aktor adalah *user* atau pengguna aplikasi yang memainkan permainan *truth or dare* menggunakan smartphone.



Deskripsi:

Pengguna yang ingin memainkan permainan truth or dare menggunakan smartphone dapat membuka dahulu aplikasi Jurani. Setelah membuka aplikasi Jurani, pengguna diharuskan terlebih dahulu memasukkan jumlah pemain yang akan bermain, kemudian pengguna memasukkan nama-nama pemain sesuai dengan jumlah pemain yang akan mengikuti permainan. Setelah itu, pengguna akan dihadapkan kepada halaman pengacakan pemain, di sini nama-nama pemain akan diacak untuk mendapatkan nama pemain yang harus menerima tantangan.

Pemain yang terpilih harus memilih jenis tantangan, yaitu jujur atau berani. Setelah memilih jenis tantangan baik itu jujur ataupun berani, pemain akan dibawa ke halaman untuk pengacakan tantangan untuk mendapatkan tantangan yang harus dilakukan oleh si pemain

(jujur atau berani), kemudian selanjutnya dilakukan pengacakan terhadap nama pemain yang akan dijadikan objek dari tantangan. Pengguna dapat memilih untuk melanjutkan permainan dengan memulainya dari pengacakan nama pemain yang akan mendapatkan tantangan, atau me-reset permainan dari awal yang dimulai dengan memasukkan jumlah pemain. Aplikasi ini juga menyediakan petunjuk bagi pengguna bagaimana caranya untuk menggunakan aplikasi Jurani.

2.2 Product Functions

This subsection of the SRS should provide a summary of the functions that the software will perform.

2.3 User Characteristics

This subsection of the SRS should describe those general characteristics of the eventual users of the product that will affect the specific requirements. (See the IEEE Guide to SRS for more details).

2.4 General Constraints

Adapun batasan-batasan dalam pengembangan aplikasi Jurani adalah sebagai berikut:

1. Hanya dapat memasukkan jumlah pemain maksimal sebanyak 20 pemain
2. Aplikasi tidak menyimpan data pemain
3. Aplikasi tidak mampu menerima *input* tantangan baru dari pemain
4. Aplikasi sudah menyediakan tantangan yang akan diberikan kepada pemain terpilih
5. Aplikasi menyediakan 30 *truth* dan 30 *dare*
6. Aplikasi ini ditargetkan untuk remaja berusia 17 tahun ke-atas
7. Aplikasi ini hanya dapat dijalankan di *platform* android dengan minimal versi android Ginger Bread 2.3
8. *Input* dari pemain hanya bisa berupa huruf
9. Pemain tidak bisa memasukkan nama yang sama ke dalam aplikasi
10. Menggunakan Bahasa Indonesia dalam setiap kalimat tantangan yang tersedia
11. Tantangan memiliki format SPO (Subyek-Predikat-Obyek)

2.5 Assumptions and Dependencies

This subsection of the SRS should list each of the factors that affect the requirements stated in the SRS. These factors are not design constraints on the software but are, rather, any changes to them that can affect the requirements in the SRS. For example, an assumption might be that a specific operating system will be available on the hardware designated for

the software product. If, in fact, the operating system is not available, the SRS would then have to change accordingly.

3. Requirements Specification

3.1 External Interface Requirements

3.1.1 User Interfaces

3.1.2 Hardware Interfaces

3.1.3 Software Interfaces

3.1.4 Communications Interfaces

3.2 Functional Requirements

This section describes specific features of the software project. If desired, some requirements may be specified in the use-case format and listed in the Use Cases Section.

3.2.1 <Functional Requirement or Feature #1>

3.2.1.1 Introduction

3.2.1.2 Inputs

3.2.1.3 Processing

3.2.1.4 Outputs

3.2.1.5 Error Handling

3.2.2 <Functional Requirement or Feature #2>

...

3.3 Use Cases

Use Case Scenario

Secara lebih mendetail, masing-masing use case yang terdapat pada diagram use case, dijabarkan dalam skenario use case. Di dalam skenario use case, akan diberikan uraian nama use case, aktor yang berhubungan dengan use case tersebut, tujuan dari use case, deskripsi global tentang use case, kondisi awal yang harus dipenuhi dan kondisi akhir yang diharapkan setelah berjalannya fungsional use case. Selain itu juga akan diberikan ulasan yang berkaitan dengan tanggapan dari sistem atas suatu aksi yang diberikan oleh aktor.

PROTOTYPE I

1. Memasukkan Jumlah Pemain



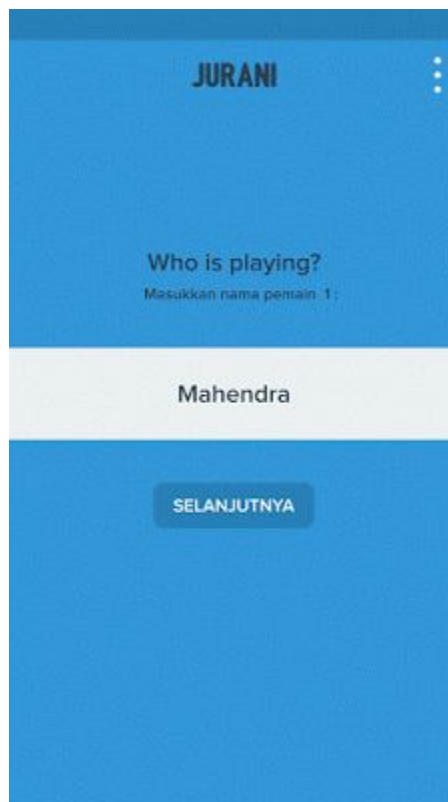
Gambar 1.1. Tampilan Halaman Memasukkan Jumlah Pemain

Primary actor: <i>User</i> yang menggunakan aplikasi Jurani.			
Description: Memasukkan jumlah pemain yang ingin mengikuti permainan.			
Policy: Minimal yang bisa ikut berpartisipasi sebanyak 2 orang dan maksimal sebanyak 20 pemain.			
Pre Condition: Pengguna membuka aplikasi Jurani dan telah melewati halaman <i>splash</i> .			
Trigger: Ingin memulai permainan Jujur atau Berani dengan versi yang lebih modern, yaitu menggunakan aplikasi <i>mobile</i> .			
Major input Jumlah Pemain	Source -	Major outputs -	Destination -

Major Step Performed

1. Membuka aplikasi Jurani
2. Memasukkan jumlah pemain

2. Memasukkan Nama Pemain



Gambar 1.2. Tampilan Halaman Memasukkan Nama Pemain

Primary actor: *User* yang menggunakan aplikasi Jurani.

Description: Memasukkan nama-nama pemain yang berpartisipasi.

Policy: Pemain hanya bisa memasukkan nama pemain berupa huruf dan bukan angka.

Pre Condition: Memasukkan Jumlah Pemain.

Trigger: Untuk mengetahui nama para pemain yang turut serta mengikuti permainan.

Major input

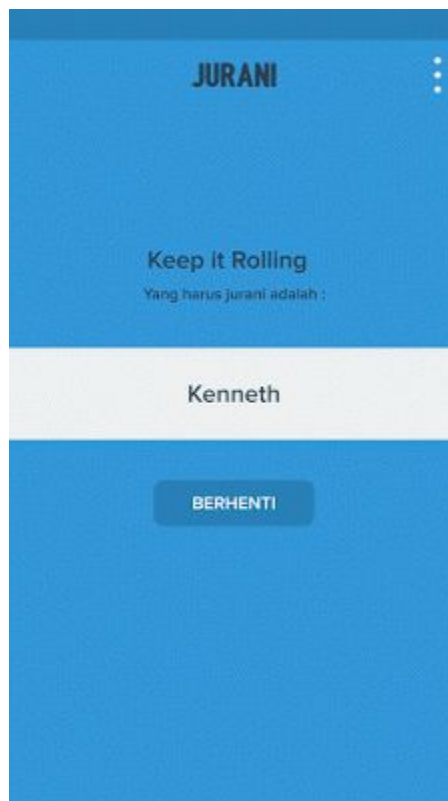
Source

Major outputs

Destination

Nama Pemain	-	-	-
Major Step Performed <ol style="list-style-type: none"> 1. Membuka aplikasi Jurani 2. Memasukkan jumlah pemain 3. Memasukkan nama-nama pemain sesuai dengan jumlah yang diinput sebelumnya. 			

3. Mengacak Nama Pemain



Gambar 1.3. Tampilan Halaman Pengacakan Nama Pemain

Primary actor: <i>User</i> yang menggunakan aplikasi Jurani.
Description: Melakukan pengacakan pemain untuk menentukan siapa yang mendapat tantangan jujur atau berani di antara <i>list</i> pemain yang berpartisipasi.
Policy: Salah satu pemain harus memberhentikan sistem yang sedang mengacak nama para pemain dengan menekan <i>button</i> “Berhenti”.
Pre Condition: Memasukkan Nama Pemain.

Trigger: Untuk mengetahui nama yang akan muncul sebagai penerima tantangan.			
Major input -	Source Nama Pemain	Major outputs Salah Satu Nama Pemain	Destination -
Major Step Performed <ol style="list-style-type: none"> 1. Membuka aplikasi Jurani 2. Memasukkan jumlah pemain 3. Memasukkan nama pemain 4. Sistem mengacak nama pemain 5. Pemain memberhentikan pengacakan dan muncul satu nama pemain yang akan ditantang 			

4. Mengacak Ulang Nama Pemain



Gambar 1.4. Tampilan Halaman Penentuan Nama yang Ditantang dan Pengacakan Ulang Nama Pemain atau Memilih Tantangan

Primary actor: *User* yang menggunakan aplikasi Jurani.

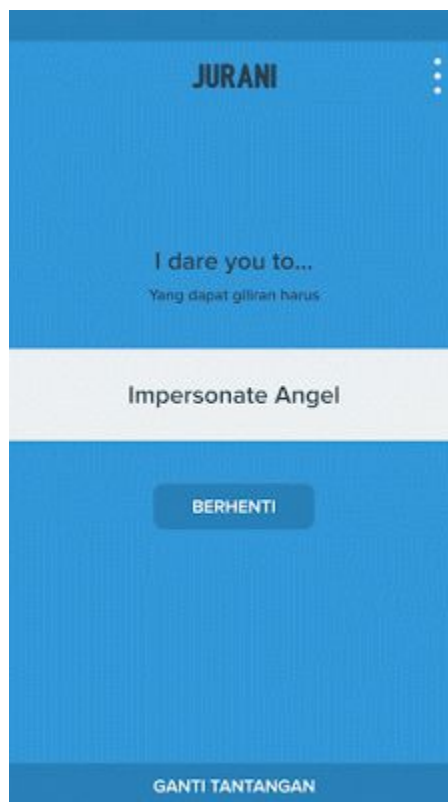
Description: Mengacak ulang list nama pemain untuk menentukan nama lainnya yang akan mendapatkan tantangan jujur atau berani.			
Policy: Pemain memilih <i>button</i> “Ulang” untuk melakukan pengacakan kembali. Meskipun begitu, masih ada kemungkinan nama yang muncul di sebelumnya akan muncul kembali pada saat memberhentikan pengacakan ulang oleh sistem.			
Pre Condition: Memberhentikan pengacakan nama pemain (muncul nama pemain yang akan ditantang).			
Trigger: Apabila pemain tersebut sudah terlalu sering menerima tantangan.			
Major input -	Source Nama Pemain	Major outputs Salah Satu Nama Pemain	Destination -
Major Step Performed <ol style="list-style-type: none"> 1. Membuka aplikasi Jurani 2. Memasukkan jumlah pemain 3. Memasukkan nama pemain 4. Sistem mengacak nama pemain 5. Pemain memberhentikan pengacakan dan muncul satu nama pemain yang akan ditantang 6. Pemain meminta sistem mengacak ulang nama pemain 			

5. Memilih Tantangan Jujur atau Berani

Primary actor: <i>User</i> yang menggunakan aplikasi Jurani.
Description: Pemain yang ditantang boleh memilih di antara dua kategori tantangan, yaitu Jujur atau Berani.
Policy: Pemain hanya boleh memilih Jujur atau Berani dengan menekan salah satu <i>button</i> yang tersedia.
Pre Condition: Memberhentikan pengacakan nama pemain (muncul nama pemain yang akan ditantang).
Trigger: Apabila pemain tersebut sudah siap untuk mendapatkan tantangan.

Major input	Source	Major outputs	Destination
Pilihan Jujur atau Berani	-	Semua List Tantangan Jujur atau Berani	-
Major Step Performed <ol style="list-style-type: none"> 1. Membuka aplikasi Jurani 2. Memasukkan jumlah pemain 3. Memasukkan nama pemain 4. Sistem mengacak nama pemain 5. Pemain memberhentikan pengacakan dan muncul satu nama pemain yang akan ditantang 6. Pemain memilih tantangan jujur atau berani 			

6. Mengacak Tantangan Jujur atau Berani



Gambar 1.5. Tampilan Halaman Pengacakan Tantangan Berani atau Mengganti Tantangan

Primary actor: *User* yang menggunakan aplikasi Jurani.

Description: Melakukan pengacakan list tantangan jujur atau berani untuk menentukan tantangan apa yang harus dilakukan (berani) atau dijawab (jujur) serta pengacakan nama pemain yang dijadikan sebagai target untuk menjadi objek yang ditantang.

Policy:

- Pemain yang ditantang harus memberhentikan sistem yang sedang mengacak list tantangan jujur atau berani dengan menekan *button* “Berhenti”.
- Apabila salah satu tantangan sudah terpilih, nama target yang ditantang sudah otomatis teracak oleh sistem. Sehingga, apabila pemain ingin mengubah nama targetnya maka tantangan pun harus diacak kembali.

Pre Condition: Memilih salah satu kategori tantangan, yaitu Jujur atau Berani agar pengacakan list tantangan disesuaikan dengan kategori tersebut.

Trigger: Untuk mengetahui tantangan yang harus diterima oleh penerima tantangan.

Major input	Source	Major outputs	Destination
-	<ul style="list-style-type: none">- Semua List Tantangan Jujur atau Berani- Semua Nama Pemain	<ul style="list-style-type: none">- Salah Satu Tantangan Jujur atau Berani- Nama Target yang Ditantang	-

Major Step Performed

1. Membuka aplikasi Jurani
2. Memasukkan jumlah pemain
3. Memasukkan nama pemain
4. Sistem mengacak nama pemain
5. Pemain memberhentikan pengacakan dan muncul satu nama pemain yang akan ditantang
6. Pemain memilih kategori tantangan, jujur atau berani
7. Sistem mengacak list tantangan sesuai kategori dan nama target sebagai objek tantangan
8. Pemain memberhentikan pengacakan dan muncul satu tantangan serta targetnya

7. Mengganti Tantangan Berani ke Tantangan Jujur atau Sebaliknya

Primary actor: <i>User</i> yang menggunakan aplikasi Jurani.			
Description: Pemain yang sebelumnya memilih satu tantangan, misalnya tantangan Berani, dapat berubah pikiran untuk pindah ke tantangan Jujur.			
Policy: Sebelum pengacakan list tantangan, pemain harus terlebih dahulu menekan <i>button</i> “Ganti Tantangan” sebelum pemain tersebut menghentikan pengacakan tantangan. Setelah hal tersebut dilakukan, maka akan diarahkan langsung ke halaman pengacakan di kategori tantangan satunya.			
Pre Condition: Sistem melakukan pengacakan list tantangan untuk kategori tantangan tertentu, misalnya tantangan Berani ke tantangan Jujur.			
Trigger: Pemain tiba-tiba berubah pikiran untuk menerima kategori tantangan satunya.			
Major input -	Source -	Major outputs - Semua List Tantangan Jujur atau Berani - Semua Target yang Ditantang	Destination -
Major Step Performed <ol style="list-style-type: none"> 1. Membuka aplikasi Jurani 2. Memasukkan jumlah pemain 3. Memasukkan nama pemain 4. Sistem mengacak nama pemain 5. Pemain memberhentikan pengacakan dan muncul satu nama pemain yang akan ditantang 6. Pemain memilih kategori tantangan, jujur atau berani 7. Sistem mengacak list tantangan sesuai kategori dan nama target sebagai objek tantangan 8. Pemain menekan <i>button</i> “Ganti Tantangan” 9. Sistem mengacak list tantangan pada kategori lainnya serta nama objek tantangan 			

8. Mengacak Ulang Tantangan



Gambar 1.6. Tampilan Halaman Penentuan Tantangan serta Targetnya dan Pengacakan Ulang List Tantangan atau Melanjutkan Permainan Jurani

Primary actor: <i>User yang menggunakan aplikasi Jurani.</i>			
Description: Melakukan pengacakan ulang list tantangan jujur atau berani untuk menentukan tantangan lain apa yang harus dilakukan (berani) atau dijawab (jujur) serta pengacakan nama pemain yang dijadikan sebagai target untuk menjadi objek yang ditantang.			
Policy: Pemain memilih <i>button</i> “Ulang” untuk melakukan pengacakan kembali. Meskipun begitu, masih ada kemungkinan tantangan yang muncul di sebelumnya akan muncul kembali pada saat memberhentikan pengacakan ulang oleh sistem.			
Pre Condition: Memberhentikan pengacakan tantangan jujur atau berani (muncul satu tantangan yang akan dihadapi).			
Trigger: Apabila tantangan tersebut sering muncul untuk beberapa putaran.			
Major input	Source	Major outputs	Destination

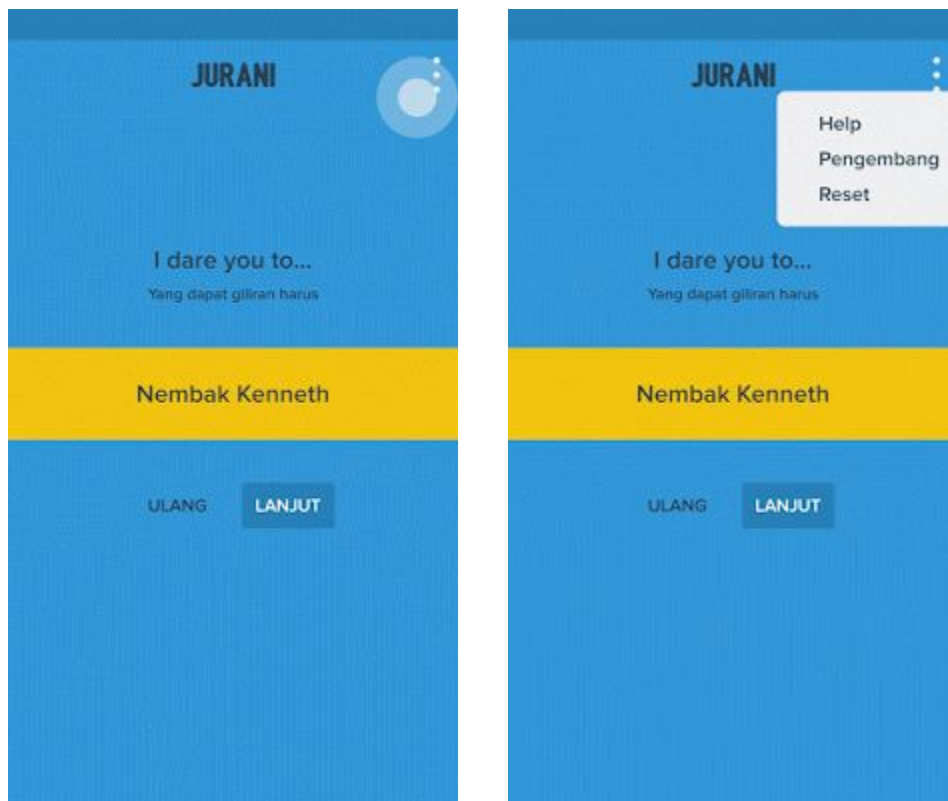
-	<ul style="list-style-type: none"> - Semua List Tantangan Jujur atau Berani - Semua Nama Pemain 	<ul style="list-style-type: none"> - Salah Satu Tantangan Jujur atau Berani - Nama Target yang Ditantang 	-
Major Step Performed <ol style="list-style-type: none"> 1. Membuka aplikasi Jurani 2. Memasukkan jumlah pemain 3. Memasukkan nama pemain 4. Sistem mengacak nama pemain 5. Pemain memberhentikan pengacakan dan muncul satu nama pemain yang akan ditantang 6. Pemain memilih kategori tantangan, jujur atau berani 7. Sistem mengacak list tantangan sesuai kategori dan nama target sebagai objek tantangan 7. Pemain memberhentikan pengacakan dan muncul satu tantangan serta targetnya 8. Pemain menekan <i>button</i> “Ulang” untuk melakukan pengacakan kembali 			

9. Melanjutkan Permainan

Primary actor: <i>User</i> yang menggunakan aplikasi Jurani.			
Description: Melakukan pengacakan kembali nama pemain untuk menentukan siapa yang akan menerima tantangan di putaran berikutnya.			
Policy: <ul style="list-style-type: none"> - Pemain menekan <i>button</i> “Selanjutnya” agar kembali lagi ke halaman pengacakan nama pemain. - Nama yang telah muncul pada putaran sebelumnya, masih ada kemungkinan bagi orang tersebut untuk merima tantangan. 			
Pre Condition: Muncul satu tantangan yang harus dihadapi serta siapa nama objeknya.			
Trigger: Para pemain sepakat untuk melanjutkan permainannya.			
Major input	Source	Major outputs	Destination

-	-	Semua Nama Pemain	-
Major Step Performed <ol style="list-style-type: none"> 1. Membuka aplikasi Jurani 2. Memasukkan jumlah pemain 3. Memasukkan nama pemain 4. Sistem mengacak nama pemain 5. Pemain memberhentikan pengacakan dan muncul satu nama pemain yang akan ditantang 6. Pemain memilih kategori tantangan, jujur atau berani 7. Sistem mengacak list tantangan sesuai kategori dan nama target sebagai objek tantangan 8. Pemain memberhentikan pengacakan dan muncul satu tantangan serta targetnya 9. Pemain menyelesaikan tantangan dan menekan <i>button</i> “Selanjutnya” 10. Ditampilkan kembali halaman pengacakan nama pemain 			

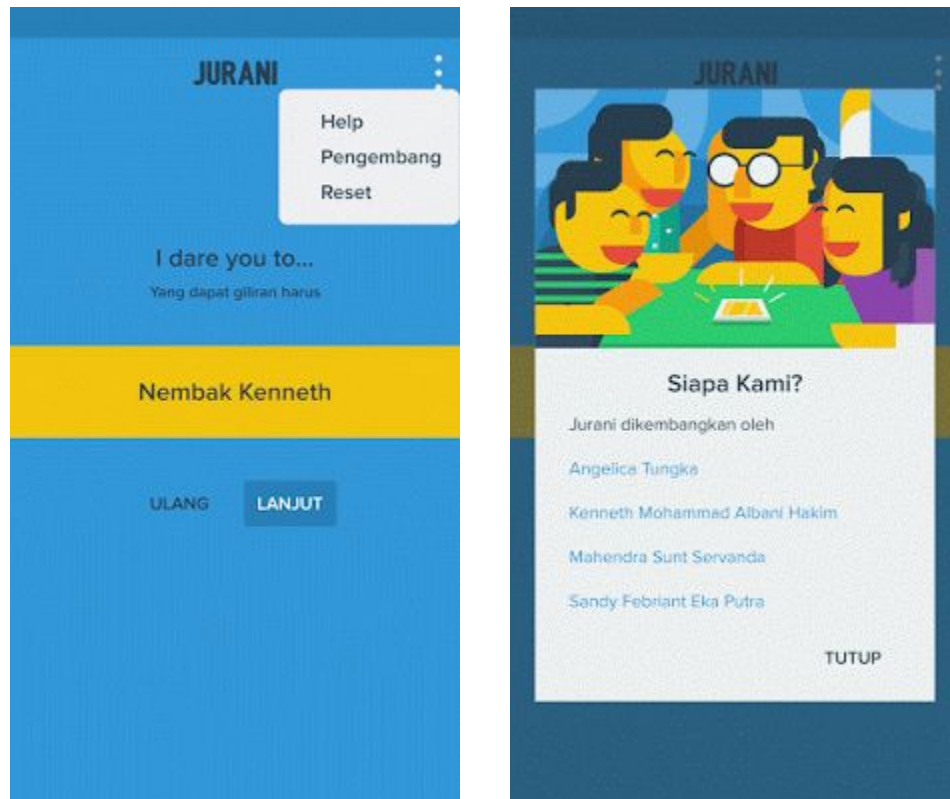
10. Reset Permainan



Gambar 1.7. Tampilan Halaman Menggunakan Fitur Reset Permainan

Primary actor: <i>User</i> yang menggunakan aplikasi Jurani.			
Description: Meng- <i>reset</i> ulang permainan seolah-olah aplikasi baru dibuka.			
Policy: <ul style="list-style-type: none"> - Menu pilihan ini ada di setiap halaman aplikasi. - Apabila pilihan <i>reset</i> ini dipilih maka akan kembali ke halaman utama, yaitu memasukkan jumlah pemain, memasukkan nama pemain, dan seterusnya. 			
Pre Condition: -			
Trigger: Pemain mengulang kembali permainan dari awal seolah-olah aplikasi ini masih <i>fresh</i> .			
Major input -	Source -	Major outputs -	Destination -
Major Step Performed <ol style="list-style-type: none"> 1. Menekan menu pilihan di pojok kanan atas 2. Memilih fitur <i>Reset</i> 3. Semua aktivitas di-<i>reset</i> oleh sistem untuk kembali ke awal dan aplikasi akan menampilkan halaman pertama permainan dimulai 			

11. Melihat Informasi Pengembang Aplikasi

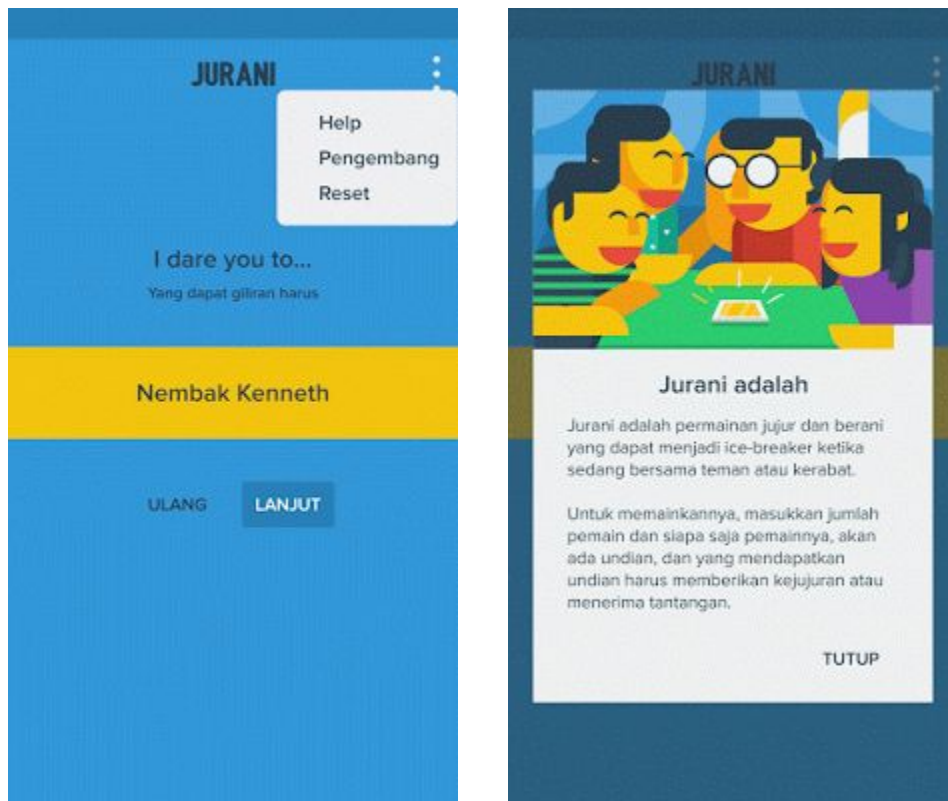


Gambar 1.8. Tampilan Halaman Melihat Informasi Pengembang Aplikasi Jurani

Primary actor: <i>User</i> yang menggunakan aplikasi Jurani.			
Description: Untuk melihat informasi seputar para pengembang aplikasi Jurani.			
Policy: <ul style="list-style-type: none"> - Menu pilihan ini ada di setiap halaman aplikasi. - Apabila pilihan ini dipilih maka tidak akan mengarah ke halaman tertentu, hanya saja ada halaman pop-up pada halaman yang sedang dibuka. 			
Pre Condition: -			
Trigger: Pemain ingin mengetahui informasi pengembang aplikasi Jurani			
Major input -	Source -	Major outputs Informasi para Pengembang Aplikasi Jurani	Destination -
Major Step Performed			

1. Menekan menu pilihan di pojok kanan atas
2. Memilih pilihan “Pengembang”
3. Muncul halaman pop-up dan pengguna dapat memilih salah satu nama yang ingin diketahui informasinya.

12. Melihat Informasi Cara Penggunaan Aplikasi



Gambar 1.9. Tampilan Halaman Melihat Informasi Seputar Jurani dan Panduan Bermain

Primary actor: *User* yang menggunakan aplikasi Jurani.

Description: Mencari tahu tentang aplikasi Jurani dan bagaimana cara memainkannya.

Policy:

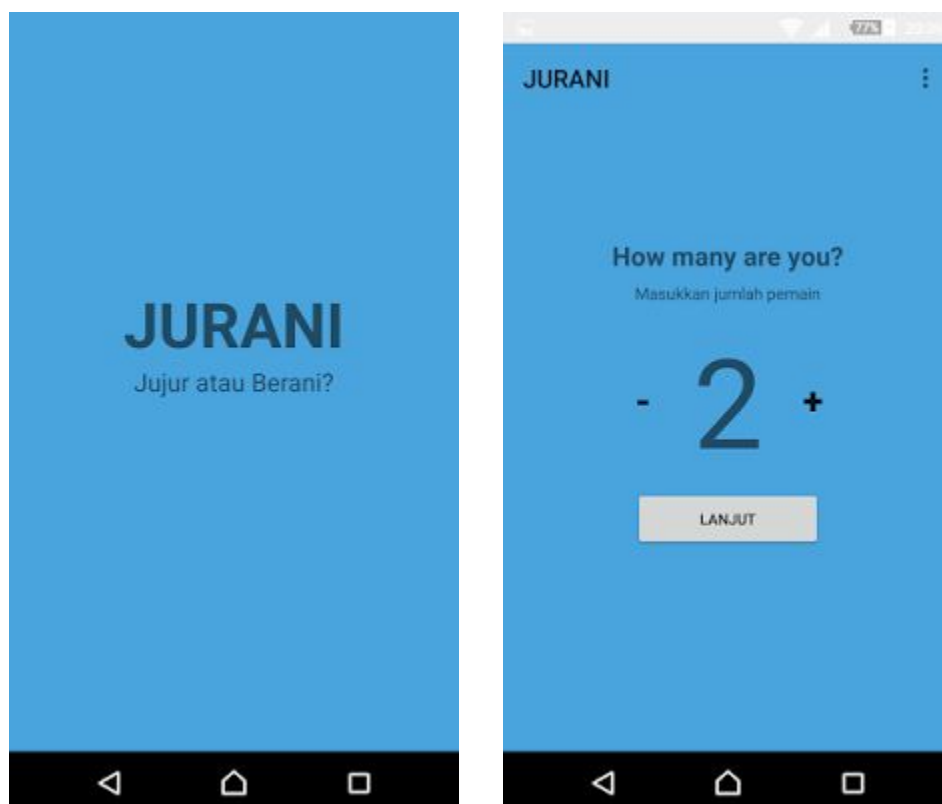
- Menu pilihan ini ada di setiap halaman aplikasi.
- Apabila pilihan ini dipilih maka tidak akan mengarah ke halaman tertentu, hanya saja ada halaman pop-up pada halaman yang sedang dibuka.

Pre Condition: -

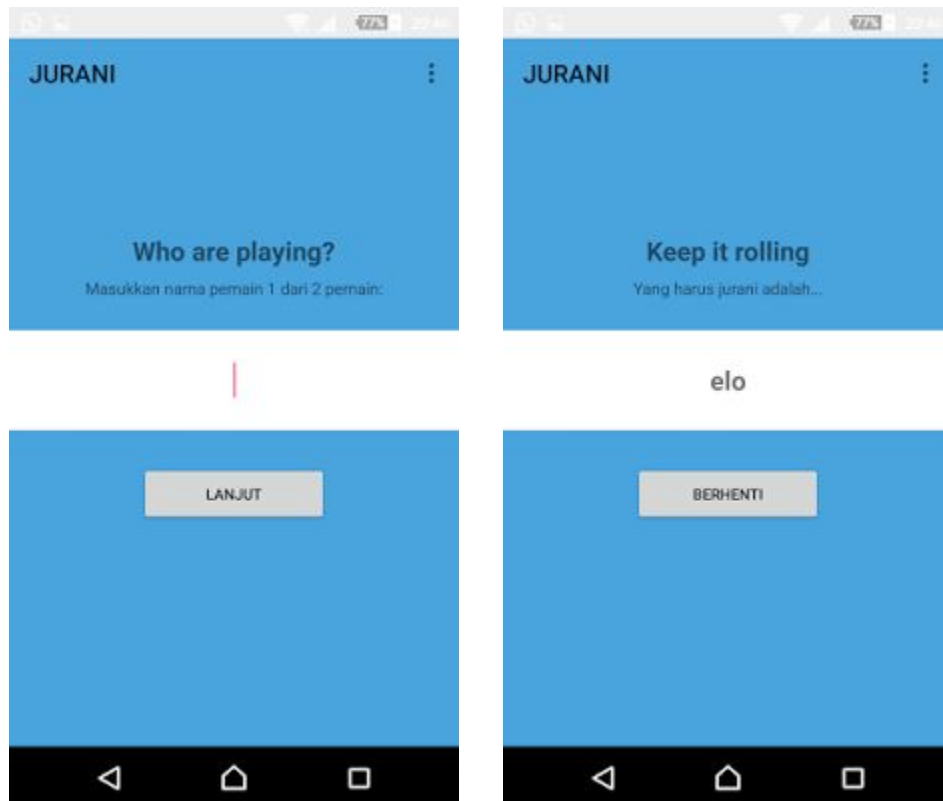
Trigger: Pemain ingin mencari tahu tentang aplikasi Jurani dan cara menggunakannya agar mereka yang memainkannya lebih antusias.			
Major input -	Source -	Major outputs Informasi Seputar Jurani	Destination -
Major Step Performed <ol style="list-style-type: none"> 1. Menekan menu pilihan di pojok kanan atas 2. Memilih fitur <i>Help</i> 3. Muncul halaman pop-up dan pengguna dapat melihat informasi tentang aplikasi Jurani dan petunjuk untuk menuntun para penggunanya. 			

Implementasi Prototype 1

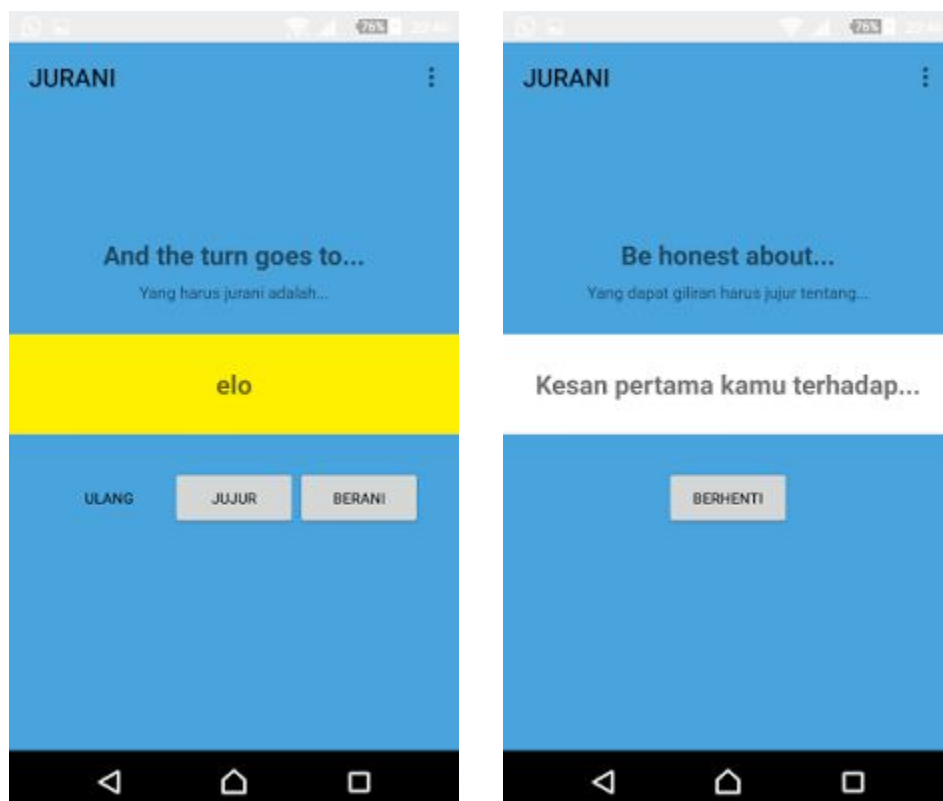
Setelah proses peng-code-an diselesaikan dengan bantuan use case scenario yang telah dibuat sebelumnya, maka proses uji coba secara internal pun dilakukan sebelum diuji oleh publik. Percobaan ini dilakukan melalui device Android.



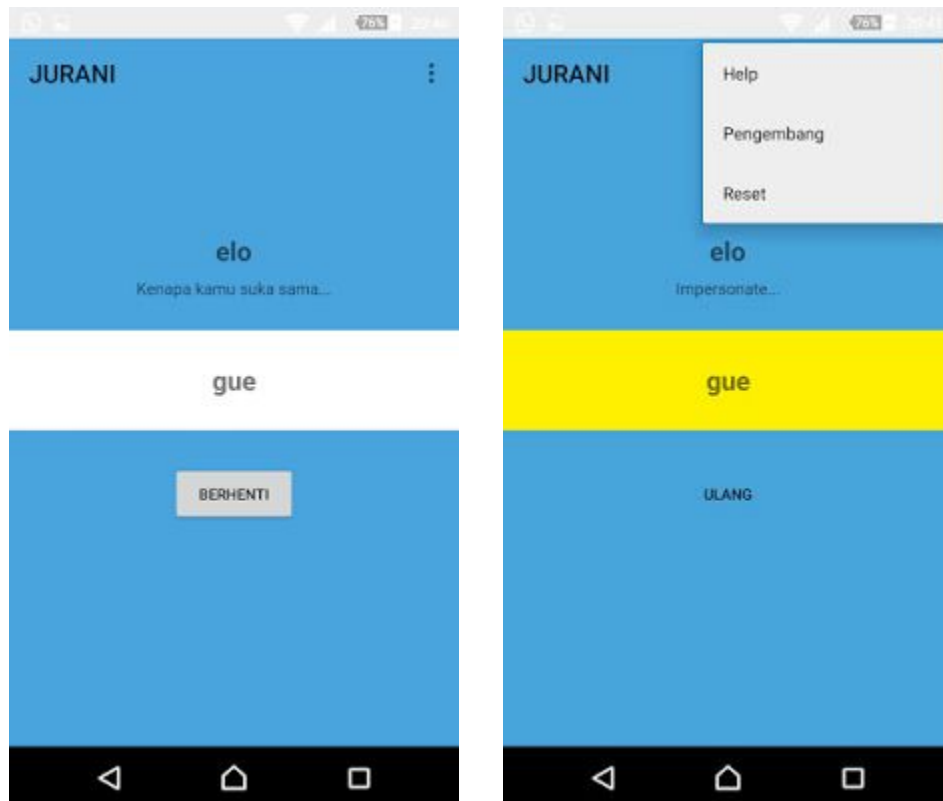
Gambar 1.10. Halaman Splash dan Memasukkan Jumlah Pemain



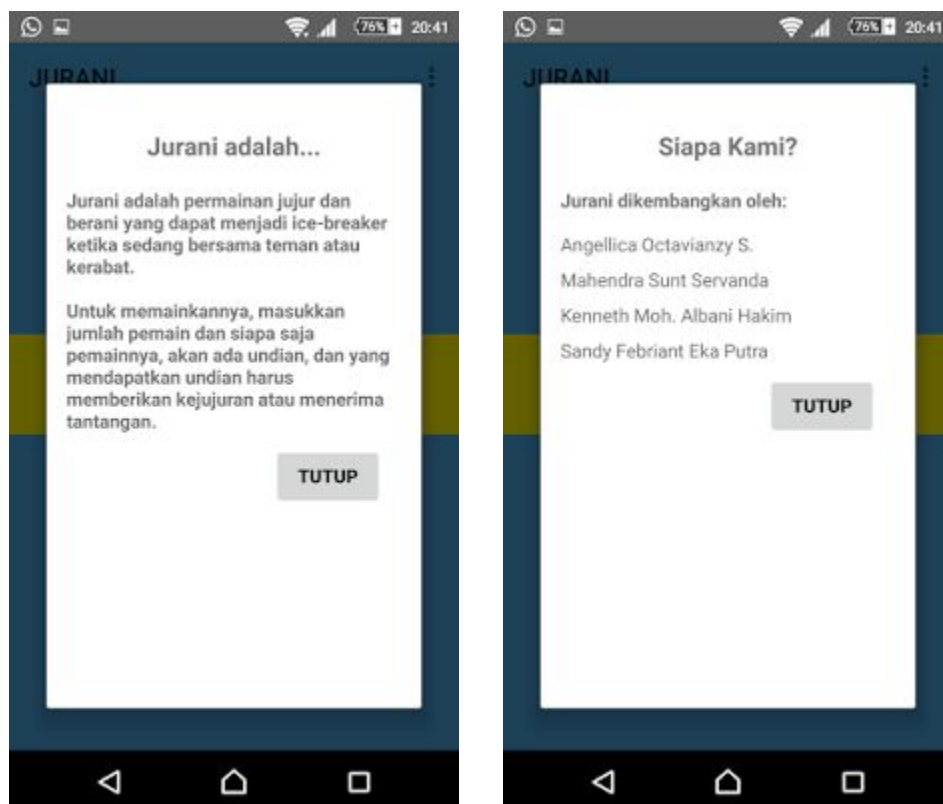
Gambar 1.11. Halaman Input dan Pengacakan Nama Pemain



Gambar 1.12. Halaman Pemilihan Tantangan dan Pengacakan Tantangan Jujur



Gambar 1.13. Halaman Penentuan Nama Objek dan Pemilihan Menu



Gambar 1.14. Halaman Informasi Jurani Beserta Panduannya dan Pengembang Aplikasi

PROTOTYPE II

Perancangan Prototype kedua ini dilakukan sesuai dengan uji coba dan saran yang diberikan oleh para pengguna. Terdapat dua kategori perubahan, yaitu dalam sisi fitur itu sendiri dan juga desain, baik *user interface* maupun *user experience*-nya. Berikut ini dipaparkan dalam bentuk user scenario untuk mempermudah proses peng-code-an.

UI dan UX

1. Icon Aplikasi Jurani

1. Pembesaran Font
2. Perbanyak Jenis Tantangan (Hingga 30 Tantangan)
3. Button “ULANG” dibuat raised button
4. Bahasa Indonesia dan Bahasa Inggris Digunakan Bercampuran
- 5.
6. *Help-nya numbering dan sekuensial*
7. *Style di status bar*

Fitur

1. Tombol BackSetiap di-”back”, memberikan pop up 'mau keluar?' ya/tidak
2. Percepat Pengacakan
3. Keyboard “Lanjut”
4. Melakukan *filter* terhadap nama yang dimasukkan, jika sudah ada nama yang sama maka pemain harus memasukkan nama yang lain
5. Dikasih *dialog* pada saat *reset* aplikasi "Apakah kamu yakin ingin me-reset permainan?"

1. Mengacak Tantangan Jujur atau Berani

Gambar 2.1. Tampilan Halaman Mengacak Tantangan Jujur

Primary actor: <i>User</i> yang menggunakan aplikasi Jurani.
Description: Melakukan pengacakan list tantangan jujur atau berani untuk menentukan tantangan apa yang harus dilakukan (berani) atau dijawab (jujur).
Policy: Pemain yang ditantang harus memberhentikan sistem yang sedang mengacak list tantangan jujur atau berani dengan menekan <i>button</i> “Berhenti”.
Pre Condition: Memilih salah satu kategori tantangan, yaitu Jujur atau Berani agar pengacakan list tantangan disesuaikan dengan kategori tersebut.

Trigger: Untuk mengetahui tantangan yang harus diterima oleh penerima tantangan.

Major input	Source	Major outputs	Destination
-	Tantangan Jujur atau Berani	Semua List Tantangan Jujur atau Berani	-

Major Step Performed

1. Membuka aplikasi Jurani
2. Memasukkan jumlah pemain
3. Memasukkan nama pemain
4. Sistem mengacak nama pemain
5. Pemain memberhentikan pengacakan dan muncul satu nama pemain yang akan ditantang
6. Pemain memilih kategori tantangan, jujur atau berani
7. Sistem mengacak list tantangan sesuai kategori
8. Pemain memberhentikan pengacakan dan muncul satu tantangan

3.4 Classes / Objects

3.4.1 <Class / Object #1>

3.4.1.1 Attributes

3.4.1.2 Functions

<Reference to functional requirements and/or use cases>

3.4.2 <Class / Object #2>

...

3.5 Non-Functional Requirements

Non-functional requirements may exist for the following attributes. Often these requirements must be achieved at a system-wide level rather than at a unit level. State the requirements in the following sections in measurable terms (e.g., 95% of transaction shall be processed in less than a second, system downtime may not exceed 1 minute per day, > 30 day MTBF value, etc).

3.5.1 Performance

3.5.2 Reliability

3.5.3 Availability

3.5.4 Security

3.5.5 Maintainability

3.5.6 Portability

3.6 Inverse Requirements

*State any *useful* inverse requirements.*

3.7 Design Constraints

Specify design constraints imposed by other standards, company policies, hardware limitation, etc. that will impact this software project.

3.8 Logical Database Requirements

Will a database be used? If so, what logical requirements exist for data formats, storage capabilities, data retention, data integrity, etc.

3.9 Other Requirements

Catchall section for any additional requirements.

4. Analysis Models

List all analysis models used in developing specific requirements previously given in this SRS. Each model should include an introduction and a narrative description. Furthermore, each model should be traceable the SRS's requirements.

4.1 Sequence Diagrams

4.3 Data Flow Diagrams (DFD)

4.2 State-Transition Diagrams (STD)

5. Change Management Process

Identify and describe the process that will be used to update the SRS, as needed, when project scope or requirements change. Who can submit changes and by what means, and how will these changes be approved.

A. Appendices

Appendices may be used to provide additional (and hopefully helpful) information. If present, the SRS should explicitly state whether the information contained within an appendix is to be considered as a part of the SRS's overall set of requirements.

Example Appendices could include (initial) conceptual documents for the software project, marketing materials, minutes of meetings with the customer(s), etc.

A.1 Appendix 1

A.2 Appendix 2

Format *Software Requirements Specifications*

- Bab 1 : Gambaran dan penjelasan ringkasan produk
- Bab 2 : Lingkungan pengembangan, pengoperasian, dan pemeliharaan
- Bab 3 : Pengantarmukaan eksternal dan aliran data : format tampilan, *user command*, DFD, kamus data
- Bab 4 : Persyaratan fungsional : fungsi-fungsi yang diinginkan
- Bab 5 : Persyaratan kinerja : tanggapan, waktu proses
- Bab 6 : Penanganan kesalahan : aksi dan pesan yang harus dilakukan sebagai tanggapan atas input atau situasi yang tidak dikehendaki produk
- Bab 7 : Subset permulaan dan prioritas implementasi : 'versi' awal produk
- Bab 8 : Perkiraan modifikasi dan peningkatan
- Bab 9 : Kriteria penerimaan
- Bab 10 : Petunjuk dan panduan perancangan
- Bab 11 : Index acuan
- Bab 12 : Daftar istilah