CPE 333 Software Engineering

Horoscope Application on iOS

Present to

Aj. Phond Phunchongharn
Aj. Khajonpong Akkarajitsakul

Created by

Gypsy

Varitta Sombunying	ID : 56070503431
Intuon Pattarakitntitkul	ID : 56070503443
Aim Manmalai	ID : 56070503444
Saran Wongvisetsak	ID : 56070503486
Supanun Kaewsri	ID : 56070503487

Department of Computer Engineering
King Mongkut's University of Technology Thonburi

CONTENTS

INTRODUCTION

Horoscope Application on iOS is used to forcast people future or another word is fortunetelling. People and fortune are come along together. When people would like some help from gods, they go pray and take some risk to forecast their future or asking some questions by shaking a fortunate-stick or etc. Actually, they just find a place to be clam and express their feeling. Hence, people do not have much time in present time so it will be good if they can find a place for praying or do online horoscope.

This application will answer those requirements. In our application we separate into 2 parts, free and purchase. Nevertheless, users have to log in via facebook, then select menu (funtions) to do next. Overall we have 9 funtions, those are; Tarot, Fourtune-Stick, recommended places, daily horoscope, phone number, prayer book, soulmate fortelling, show result in graph, and share result on facebook. For purchased functions, you have to buy coins by electronic payment that support out application.

Although you have no time to go out, you can still enjoy the forcast you like, our application provides you recommend places or temples, and also horoscope stuff you may interested.

Regards,
Developer Team

- PURPOSE OF THE DESIGN DOCUMENT.

The challenge of this project is a new programming language that we have to learn by ourselves. It is called Swift, this language is supported on Apple devices. It needs Xcode to generate and execute. Our goal is to finish this application as soon as we can, and the user interface has to be nice too. Even though we had planned to finish the application, but it also depends on other components and time.

- DOCUMENT CONVENTIONS.

Definitions of all term in application.

Term	Definition
Horoscope	Application Name (Name of product)
User	Who interacts with application
Menu	Main menu will show menu icon for users to select
Developer	Who programs and develops this application.
Partner	People who join our application with terms&conditions we both agree. (Fortune teller & Developer)

- PROJECT BACKGROUND AND SCOPE.

Horoscope is an application for fortunetelling in iOS application. It is free to download from App store.

- Users can choose type of fortunetelling from menu.
- Users can share prediction on facebook.
- Users can booking the time to meet the seer.
- If users got bad prediction, this application will show where and how to remove bad luck. It links with google map for recommended places.
- Users can set popup to show prediction in notification everyday.
- The prediction can show in graph.
- If users want to use full function from this application, users must purchase for application.
- Fortune teller can join to be a part of our application by signing up with terms & conditions that we have provided. They have to pay to advertise themselves. And developers will provide their information that users can contact them directly

- DOCUMENT OVERVIEW

This document, the Software Design Document (SDD) are requirements in the form of a task and system object model. This model shows about process of plan to make this project
This document has 2 sections, 6 topics and 9 subtopics:

- Software Design Document (SDD) Template

Introduction

-Purpose of the Design Document describbe the architecture and system design

-Document Conventions describe all terms of acronyms used in document

-Project background and scope describe about the software will do, what is the benefit for users

Data/Class Design

-Internal Software Data Structure describe about structures that passed among component

-Global Data Structure describe about structure that available to major portions of architecture

-Temporary Data Structure create for temporary

-Database Description database that use in this application

Architecture Design

-Architecture Design (LEVEL I) Create a modular program structure showing the relationships between modules to obtain the entire system's functionality

- Process Manual Specifications

Project Plan and Monitoring Method describe how to plan for the project's schedule and keep updating the plan.

Employee Work/Task Assignment Process describe a feature to collect data for project monitoring.

Final Project Cost Method With Example can determine the projects direct and indirect cost

DATA / CLASS DESIGN

- INTERNAL SOFTWARE DATA STRUCTURE

  Our Application will store a data in database. Adding and updating are not allow for users.

- GLOBAL DATA STRUCTURE

  Users can share a result to social network.

- TEMPORARY DATA STRUCTURE.

We do not have any temporary data structure. All information and data have been stored in the database.

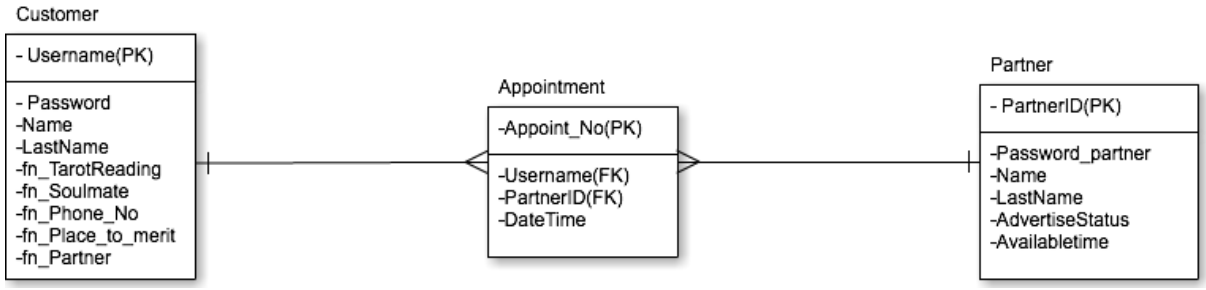
DATABASE DESCRIPTION

Table Name: Customer
Attributes: Username, Password, Name, LastName, fn_TarotReading, fn_Soulmate, fn_Phone_No, fn_Place_to_merit, fn_Partner
Description: This table would hold information of customer who use this application in the limit that they allow. A primary key of this table is “Username”.

Table Name: Appointment
Attributes: Appoint_No, Username, PartnerID,DateTime
Description: This table holds all the current appointments of customer with partner. It holds the date and time of the appointment. Primary key of this table is “Appoint_No” .

Table Name: Partner
Attributes: PartnerID, Password_Partner, Name, LastName, AdvertiseStatus, AvailableTime
Description: This table will hold the information of partner, collcet package of advertise that they was bought and shcedule of forecast. Primary key of this table is “PartnerID”.

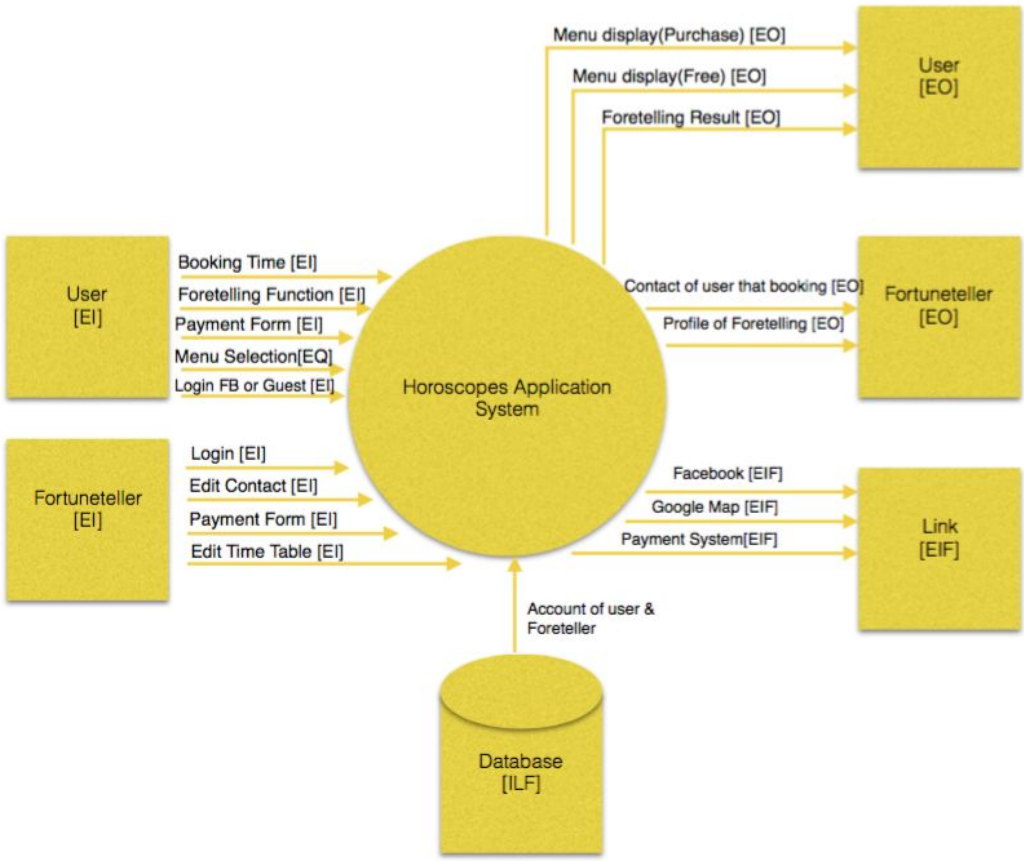
Customer	Partner	Appointment
Username(PK)	PartnerID (PK)	Appoint_No(PK)
password	Password_Partner	Username(FK)
Name	Name	PartnerID(FK)
LastName	LastName	DateTime
fn_TarotReading	AdvertiseStatus	
fn_Soulmate	AvailableTime	
fn_Phone_No	paymentStatus	
fn_Place_to_merit		
fn_Partner		
paymentStatus		


Customer
- Username(PK)
- Password
-Name
-LastName
-fn_TarotReading
-fn_Soulmate
-fn_Phone_No
-fn_Place_to_merit
-fn_Partner
Appointment
-Appoint_No(PK)
-Username(FK)
-PartnerID(FK)
-DateTime
Partner
- PartnerID(PK)
-Password_partner
-Name
-LastName
-AdvertiseStatus
-Availabletime

ARCHITECTURAL DESIGN

- ARCHITECTURAL DESIGN

Architectural Context Diagram



Description :

For the above diagram, it's describe about permission and overall of process in application.

Users

Menu display (for purchasing and free) and the result of forecast are the user can see in application.

What the user can do in app is Login with Facebook or guest, choose menu, purchase more functions, function for fortune and booking time.

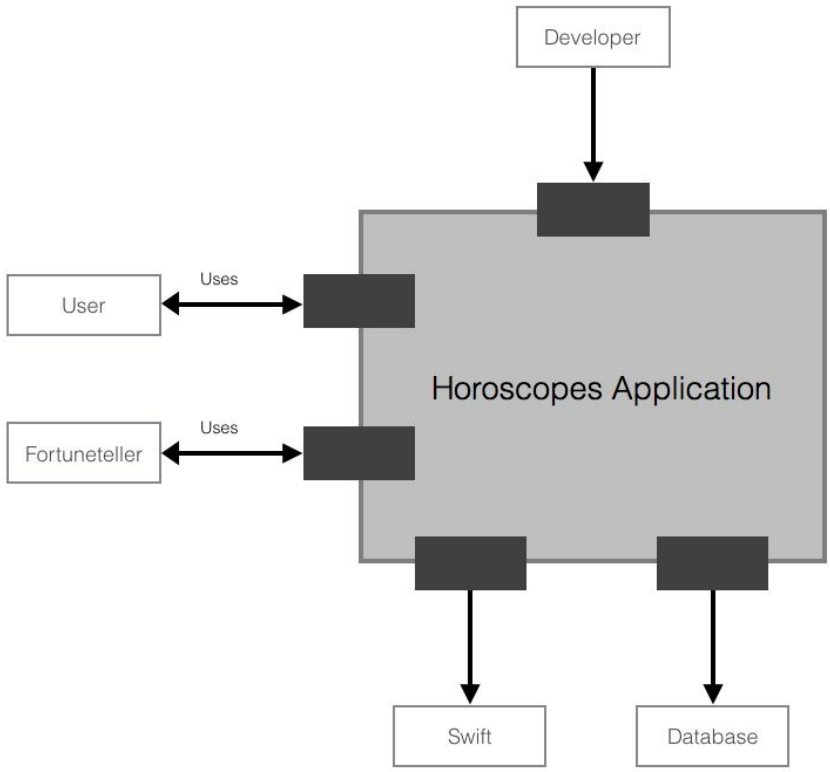
Fortunteller

The application would show profile of fortuntelling and contact of user was make appoint (schedule) on the screen.

The fortuneteller can do in the application Login, edit contact, purchase advertising and edit appointment time

This application would link with others website or socail media that is Facebook, Google Map and Payment System. We design this app to have database to collect data about information of users and fortuneteller, and scheduling about appointment between users and fortuneteller.

- UML Architectural Context Design

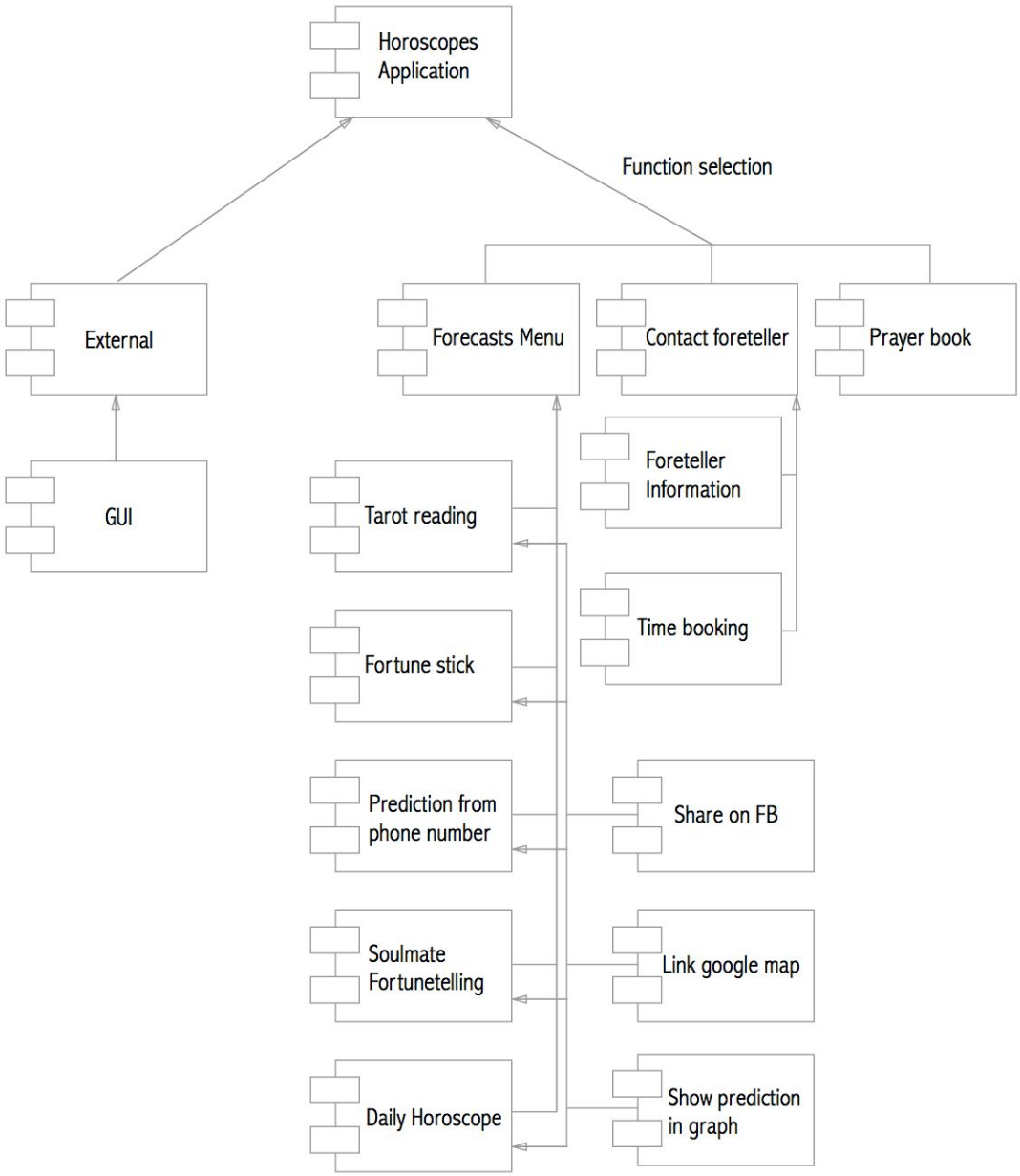


Description:

The above diagram was show about structure of application how to produce this app, target of app and who make this app.

The developer will make the app by using Swift language on platform ios and use database to collect the important and information needed. The target of user is fortunteller and person who believe or love to know their future(forecast)

- UML Component Structural Diagram



Description

From the above UML show about the detail of process in application. This divide in 2 things. First, GUI is interface that show all the data that have in this app. The next one is function in app have. If you are the new user, you have to choose how to use app between login with Facebook or guest. After that it would show the forecasts menu, contact fortuneteller and prayer book. The forecasts menu we have more functions is providing Tarot reading(Purcahse), fortune stick, prediction from phone number(Purchse), soulmate fortunetelling(Purchase) and daily horoscope, and you can see the result by text or graph. If your fortune is not good, this app would be suggest the place to excorcise. For the option to suggest place to have to purchase. This app provide to share on Facebook if you prefer.

Contact fortuneteller(Purchase) you can see the information of fortuneteller and selcet one person who you want to make an appointment. The last function in app is prayer book, you can download or watch the prayer.

COMPONENT-LEVEL DESIGN *(design 2)*

- DESCRIPTION COMPONENT

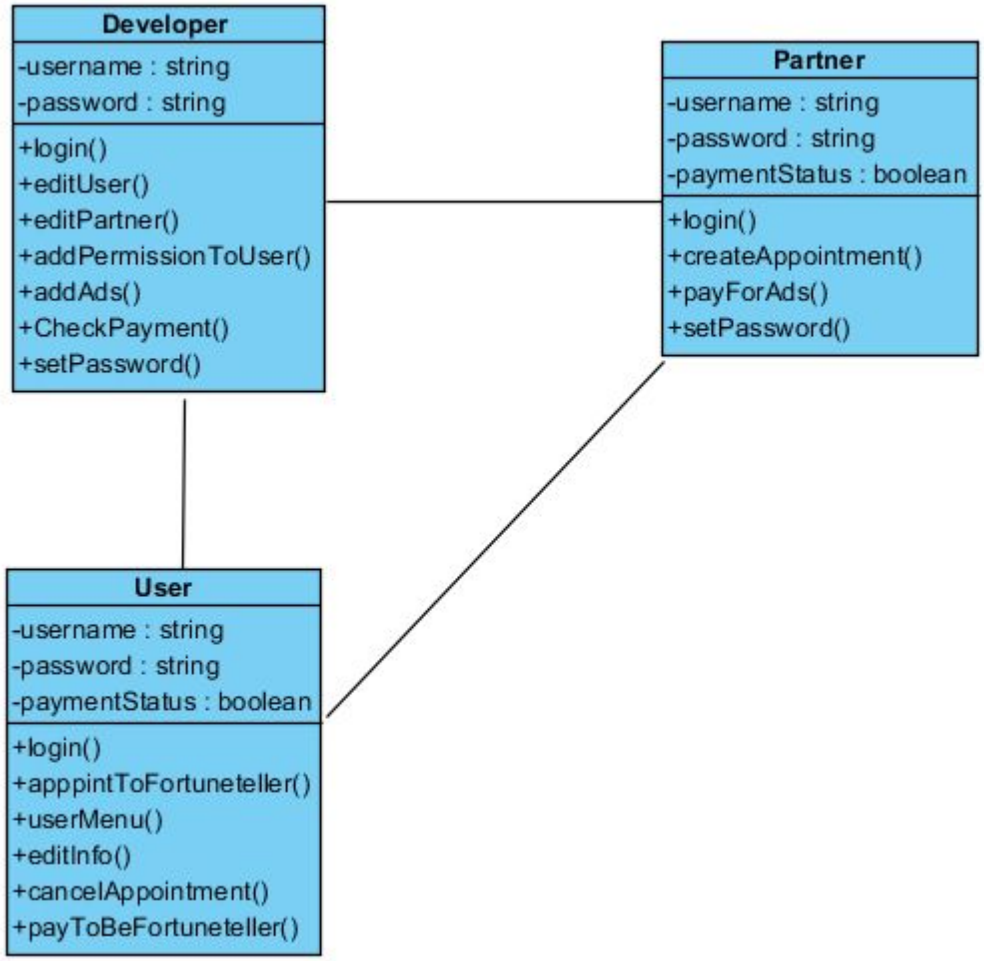
  ADMIN

  - PROCESSING NARRATIVE (PSPEC) FOR COMPONENT

The component ‘Admin’ another word is developer. It has a responsibility to take care of an application and develop better things to the application. They have permission to access an application to check and authenticate users and partners. Decide terms and conditions. Moreover, payment status of users and partners.

  - COMPONENT 1 PROCESSING DETAIL

    - Design Class hierarchy for admin



- Restrictions/limitations for component n

To access as developer, developers have to verify themselves by username and password. And developer can manage data of users, but cannot change their personal information.

- Performance issues for component n

This component have to verify when log in by admin.

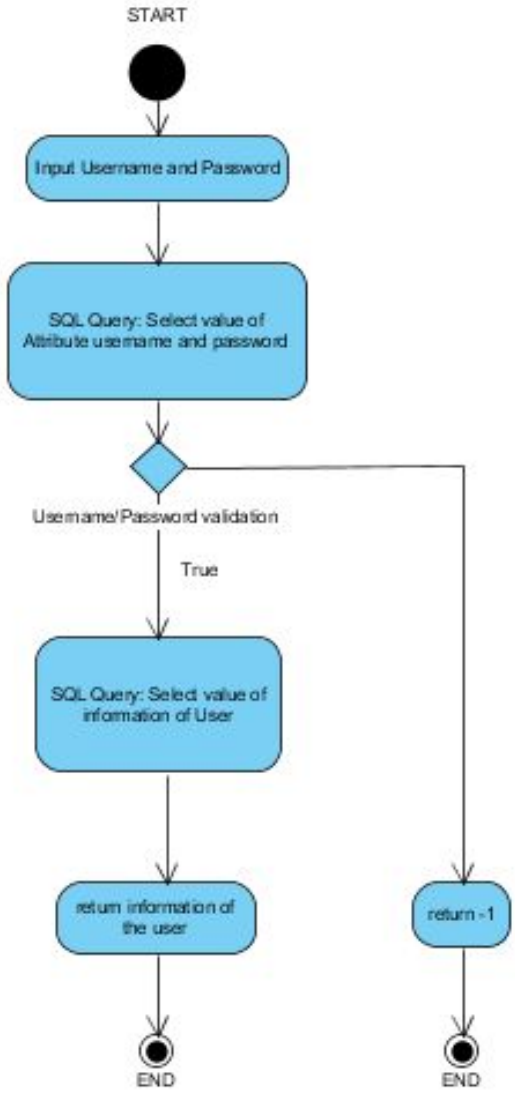
- Design constraints for component n

This component will pop up an error message for error log in

- Processing detail for each operation of component n

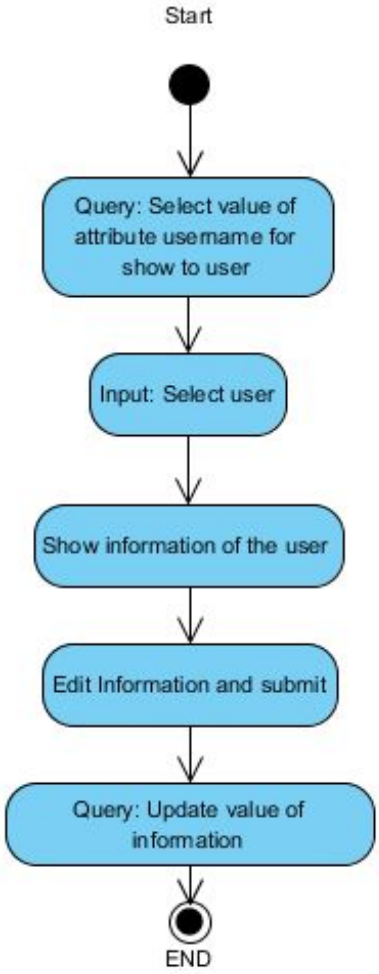
- login()

This function is used to log in to an application by checking user's password in system. If there is matched password, system will return user's data. If not system will return -1.



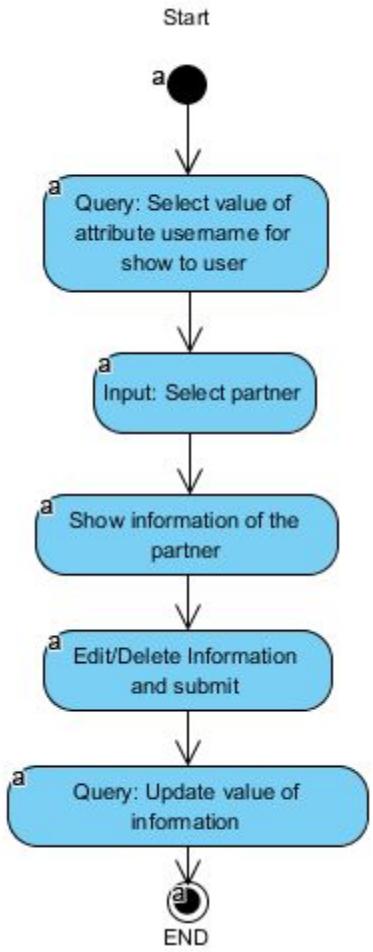
- editUser() - (for admin)

This fucntion is use to edit/delete user and user 's information.


Start
Query: Select value of attribute username for show to user
Input: Select user
Show information of the user
Edit Information and submit
Query: Update value of information
END

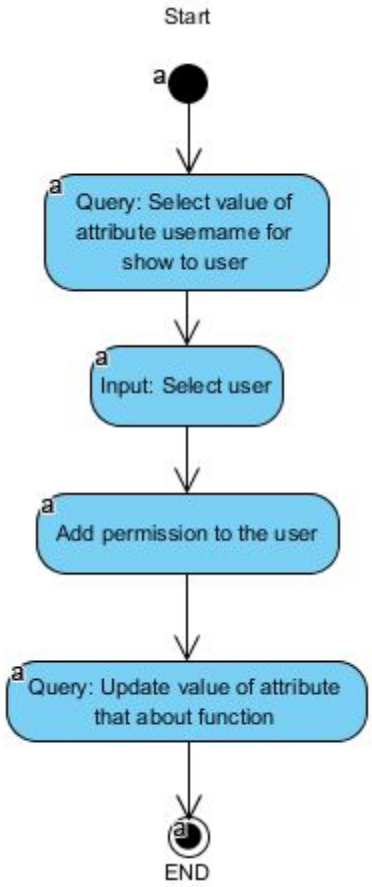
❑ editPartner() - (for admin)

This function will edit and delete partners information.

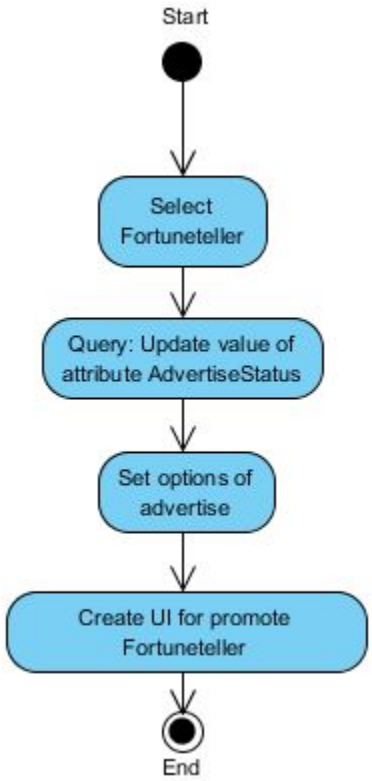


- addPermissionToUser() -(for admin)

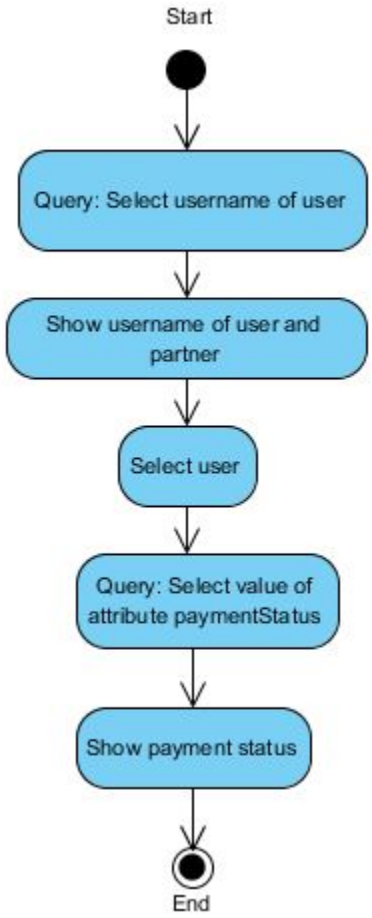
  This function is use to allow user to play application which need to purchase.



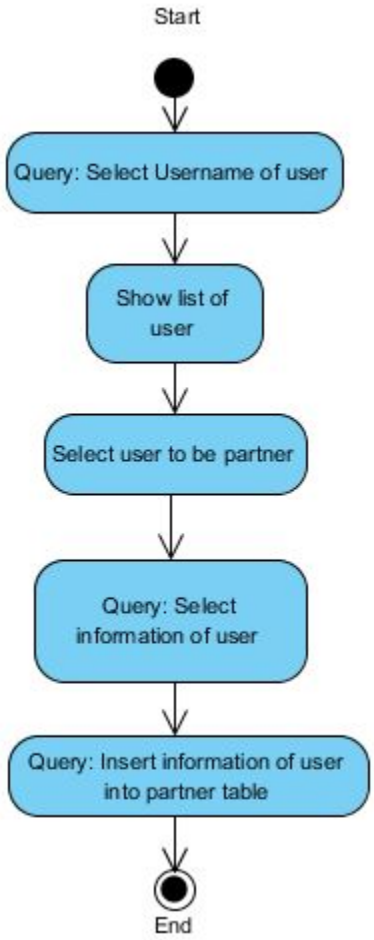
❑ addAds() - (for Admin)



❑ checkPayment() - (for Admin)



❑ addPartner() - (For admin)



- Processing narrative (PSPEC) for each operation

Admin

Process	Narrative
setPassword()	Use to set own password
addPermission()	Use to enable perrmission to user ‘s application using
ban()	Use to delete user from application
login()	Use to login to application
authenticate()	Use to authenticate users and partners(permission, payment)
addAds()	Use to add advertise for promote fortuneteller.
editUser()	Use to edit information of user and fortuneteller
checkPayment()	Use to check status of payment

- Algorithmic model (e.g., PDL) for each operation

For login

Start

Enter username and password
login to the app

End

For ban

Start

Select the account to ban
Send Banning Instruction
Receive Ban Result

End

For setPassword

Start

input old password
Input new password
Update information
Receive Result

End

For addPermission

Start

Select user to enable permission
Update information of permission
Receive Result

End

For authenticate

Start

Check value from database

Receive authenticate result

End

For addAds

Start

Select Fortuneteller

Set options

Create UI to promote fortuneteller

Receive Result

End

For editUser

Start

Select user

Recieve user's information

Edit information and submit

Receive Result

End

For checkPayment

Start

Receive list of user

Select user

Receive status of payment of user

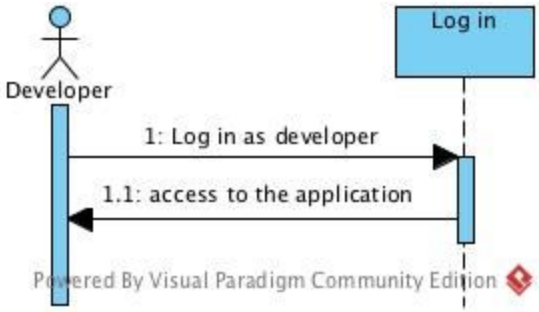
End

- COMPONENT 1 TEST POINTS LIST AND DESCRIPTION

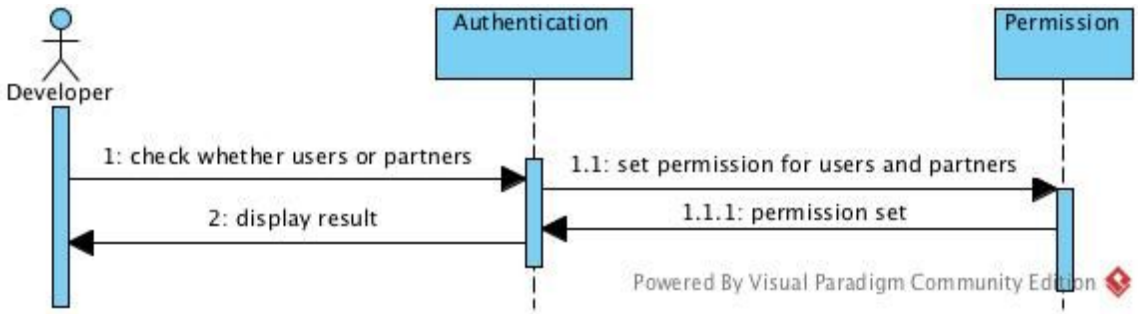
Test	Description
Fill ID and password	ID and password have to exist
Change password	Set new password
Add fortuneteller	Can add fortuneteller
Add advertise	Choose profile of fortuneteller who buy advertise to promote
Information in application	Don’t have any error or blank UI

- COMPONENT 1 DYNAMIC BEHAVIOR

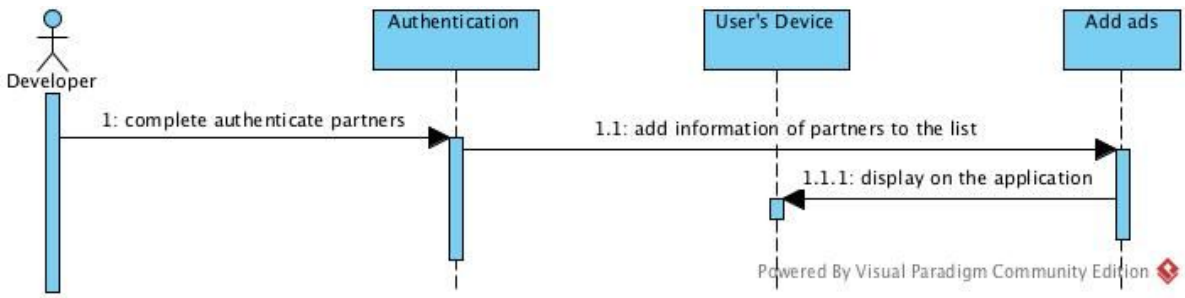
Log in as developer



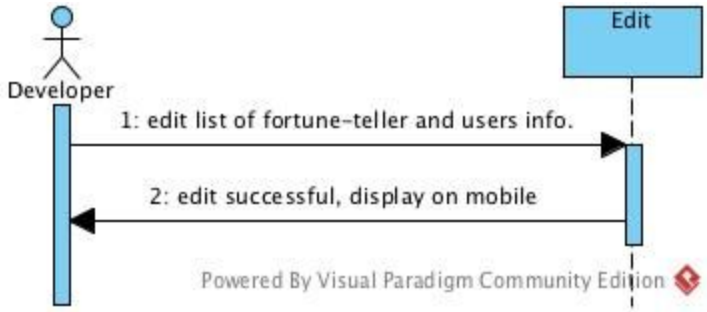
Authentication



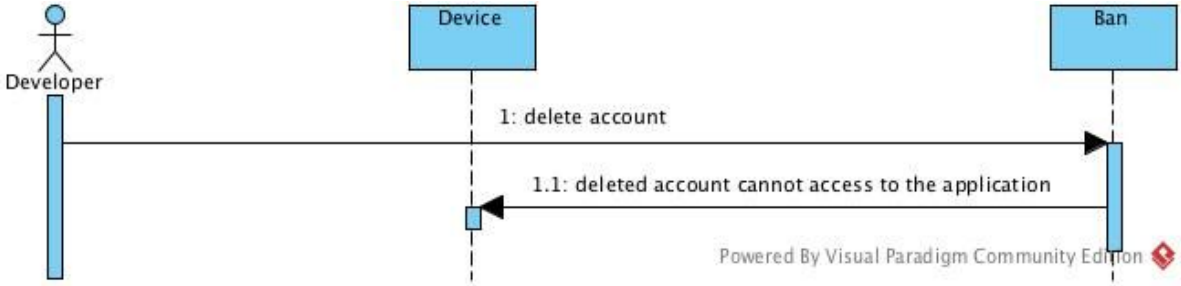
Add ads



Edit Users



Ban



- DESCRIPTION COMPONENT

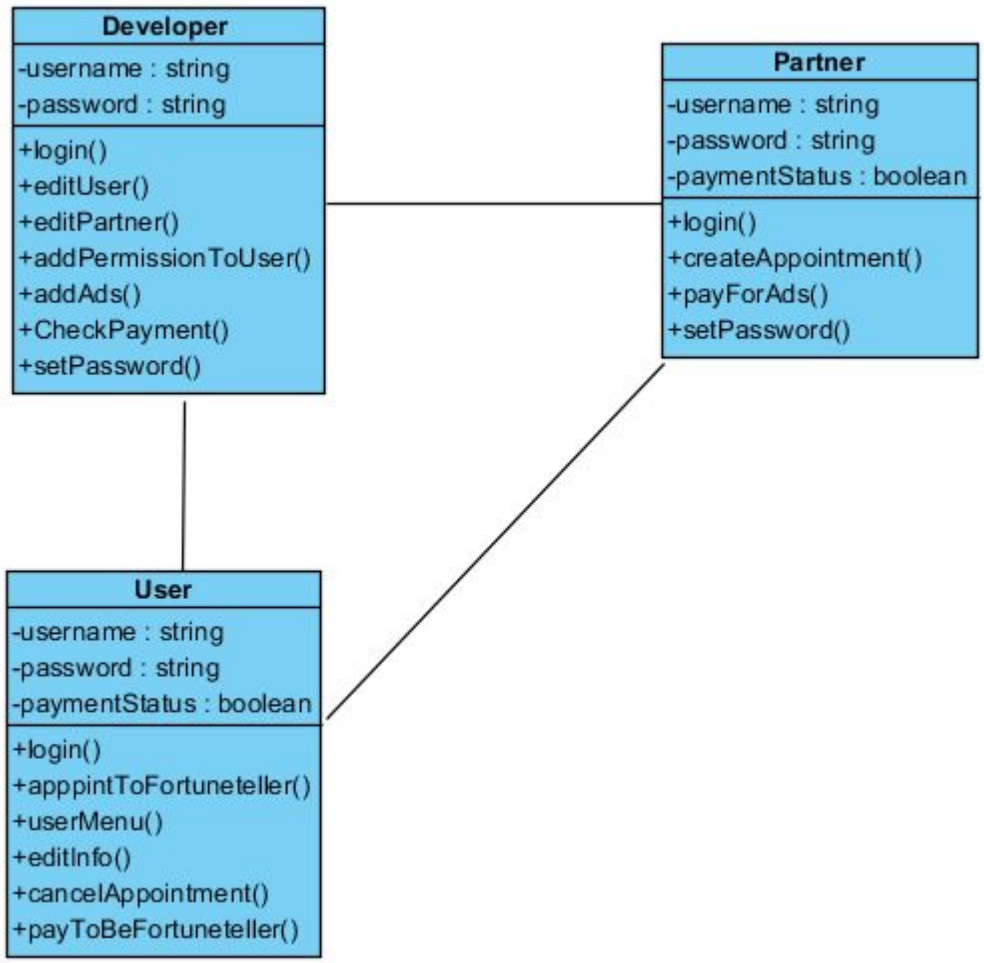
  PARTNER

  - PROCESSING NARRATIVE (PSPEC) FOR COMPONENT

The component ‘partner’ another word is fortune-teller. This component has to be authenticated by developer to join the application as partner. It requires payment system for adding partners to the list.

  - COMPONENT 1 PROCESSING DETAIL

    - Design Class hierarchy for partner



- Restrictions/limitations for component n

Partners has to contact to developer team, agree on terms and conditions of being partners, and also pay to joy the application.

- Performance issues for component n

The Component have to verify that it was log in by a partner.

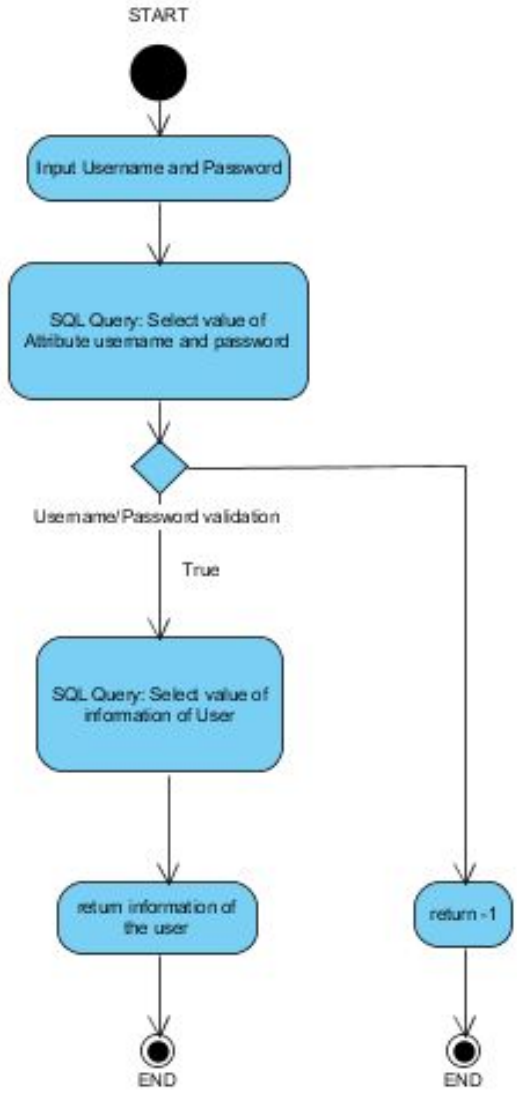
- Design constraints for component n

If the system is not loging in as a partner. It will show an error message for any case of mistaken for logging in and partner will not access a member site of appilcation.

- Processing detail for each operation of component n

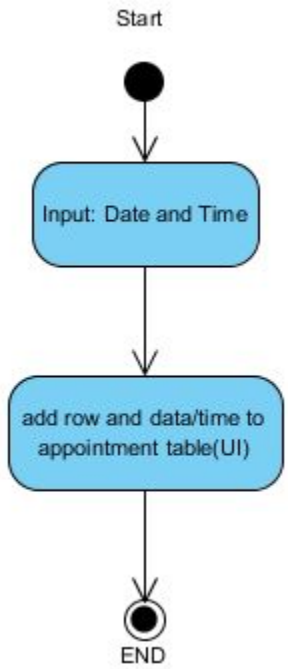
❏ login()

This function is used to log in to an application by checking user’s password in system. If there is matched password, system will return user’s data. If not system will return -1.


START
Input Username and Password
SQL Query: Select value of Attribute username and password
Username/Password validation
True
SQL Query: Select value of information of User
return information of the user
return -1
END
END

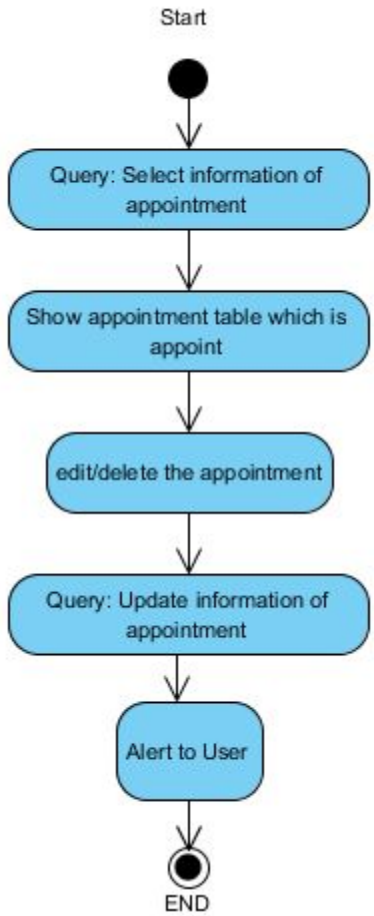
- createAppointment() - (for Partner)

  This function is use to create appointment row including date and time in appointment table which is use to appoint to fortuneteller by user.



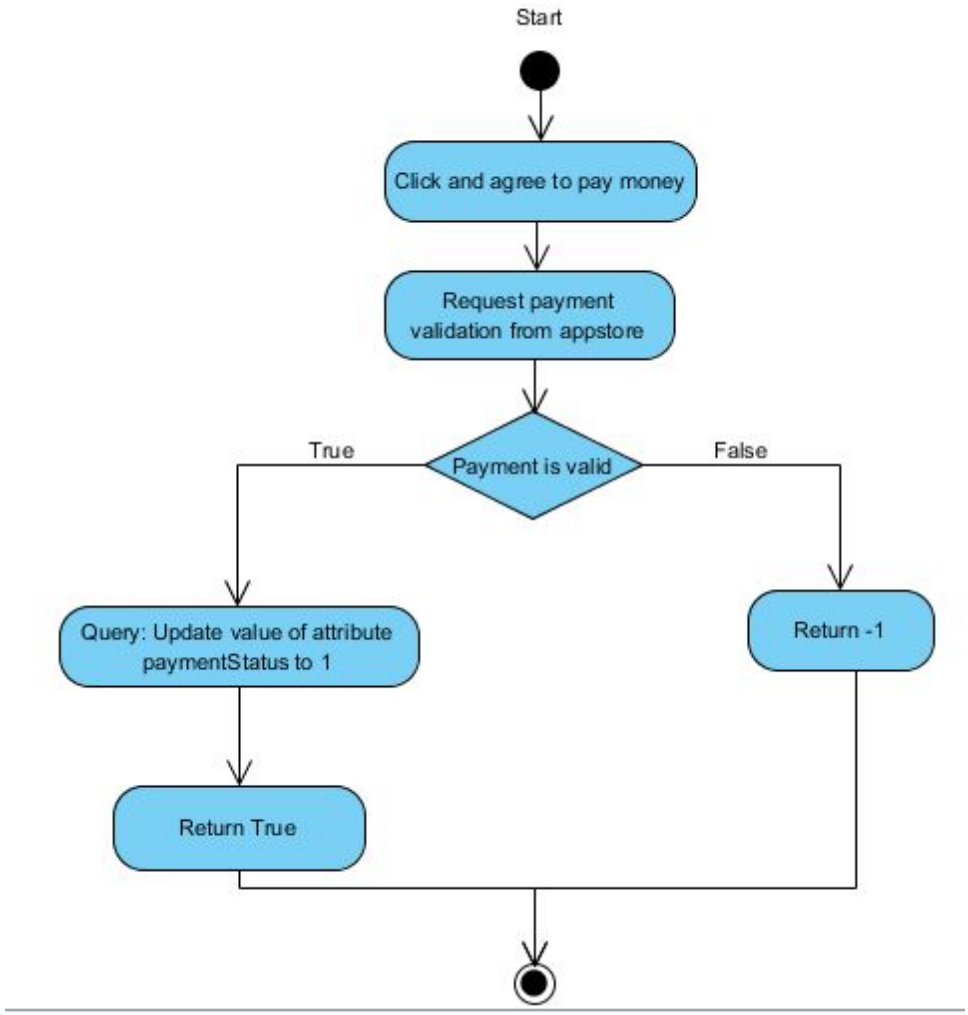
❏ editAppointment() - (for Partner)

This function is use to view appointment which is appoint with user.



- payForAds() - (for partner)

  This function is use to pay money to application for promote fortuneteller.



- Processing narrative (PSPEC) for each operation

Partner

Process	Narrative
setPassword()	Use to set own password
payForAds()	Use to request admin for advertise
login()	Use to login to application and check authentication
createAppointment()	This function is use to create appointment row including date and time in appointment table which is use to appoint to fortuneteller by user.
editAppointmentTable()	Use to edit appointment table

- Algorithmic model (e.g., PDL) for each operation

For login

Start

Enter username and password

Login to the app

End

For setPassword

Start

Input old password

Input new password

Update information

Receive Result

End

For payForAds

Start

Submit to pay money

Check payment validation

Update information of advertise

Alert to Admin

Receive Result

End

For createAppointment

Start

Input date and time
Add appointment to appointment table(UI)
Receive Result

End

For editAppointmentTable

Start

Receive appointment table
edit appointment table
Receieve Result

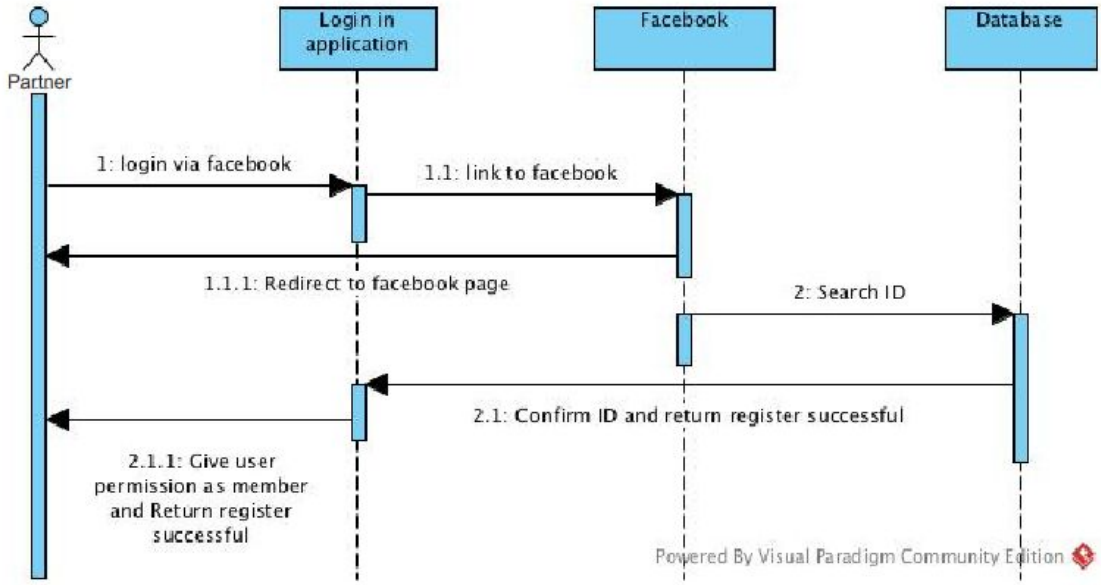
End

- COMPONENT 1 TEST POINTS LIST AND DESCRIPTION

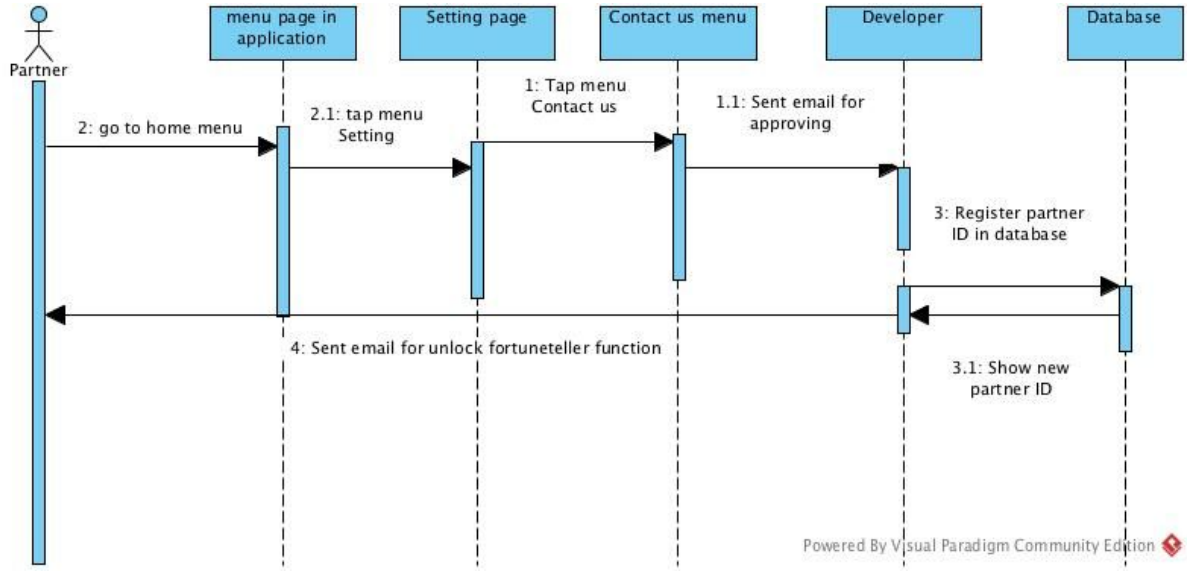
Test	Description
Facebook login	ID and password of Facebook have to exist
Change password	Set new password
Purchase for advertising	Card number and money in Credit Card have to exist
Information in application	Don’t have any error or blank UI

- COMPONENT 1 DYNAMIC BEHAVIOR

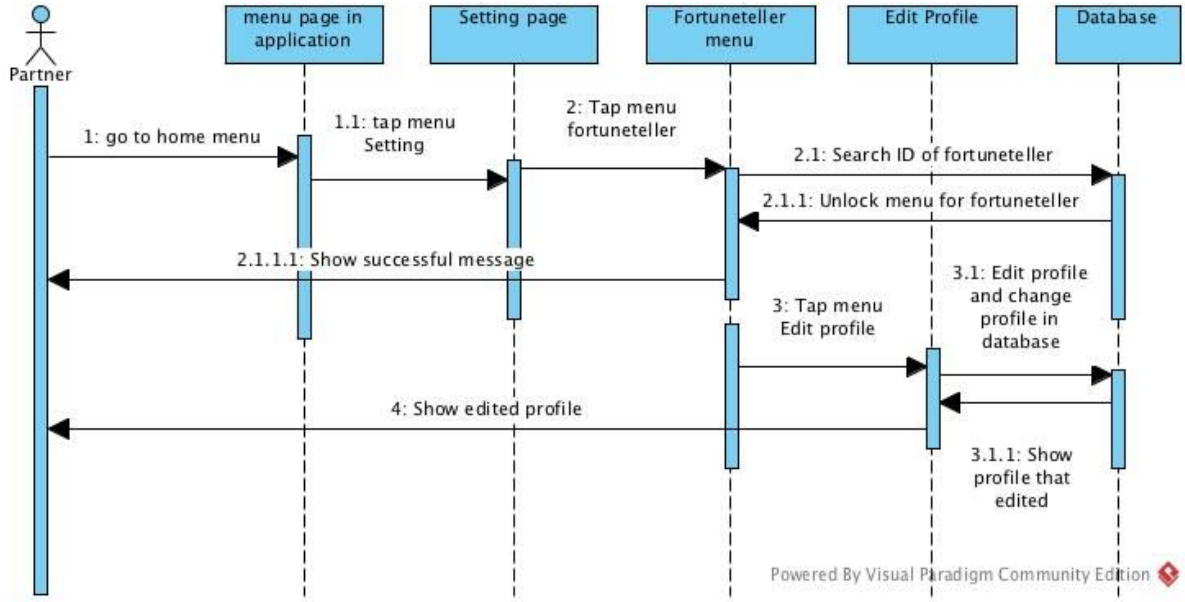
Log in via facebook



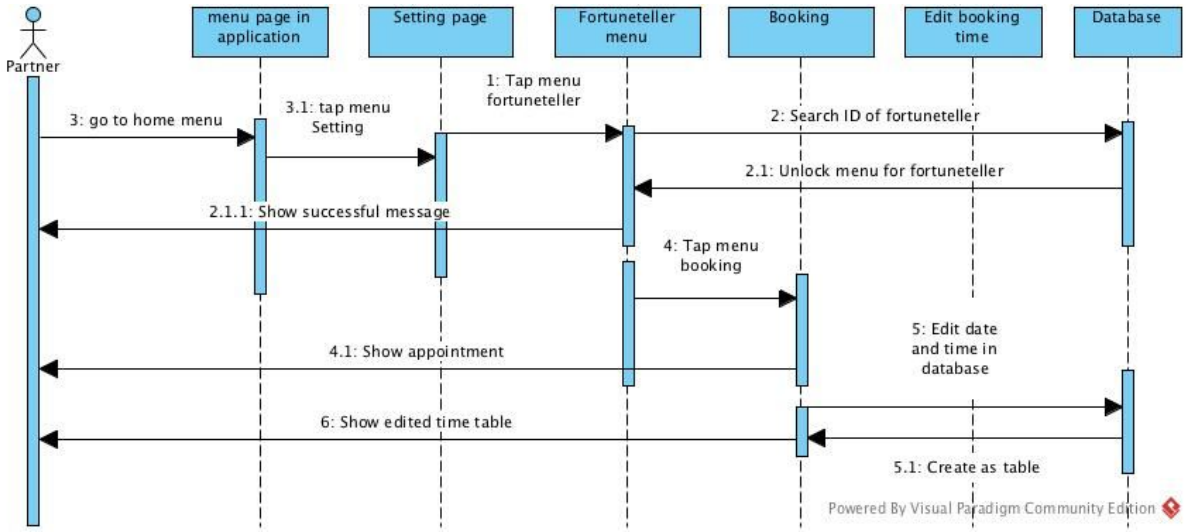
Approve



Edit proflie



Booking



- DESCRIPTION COMPONENT

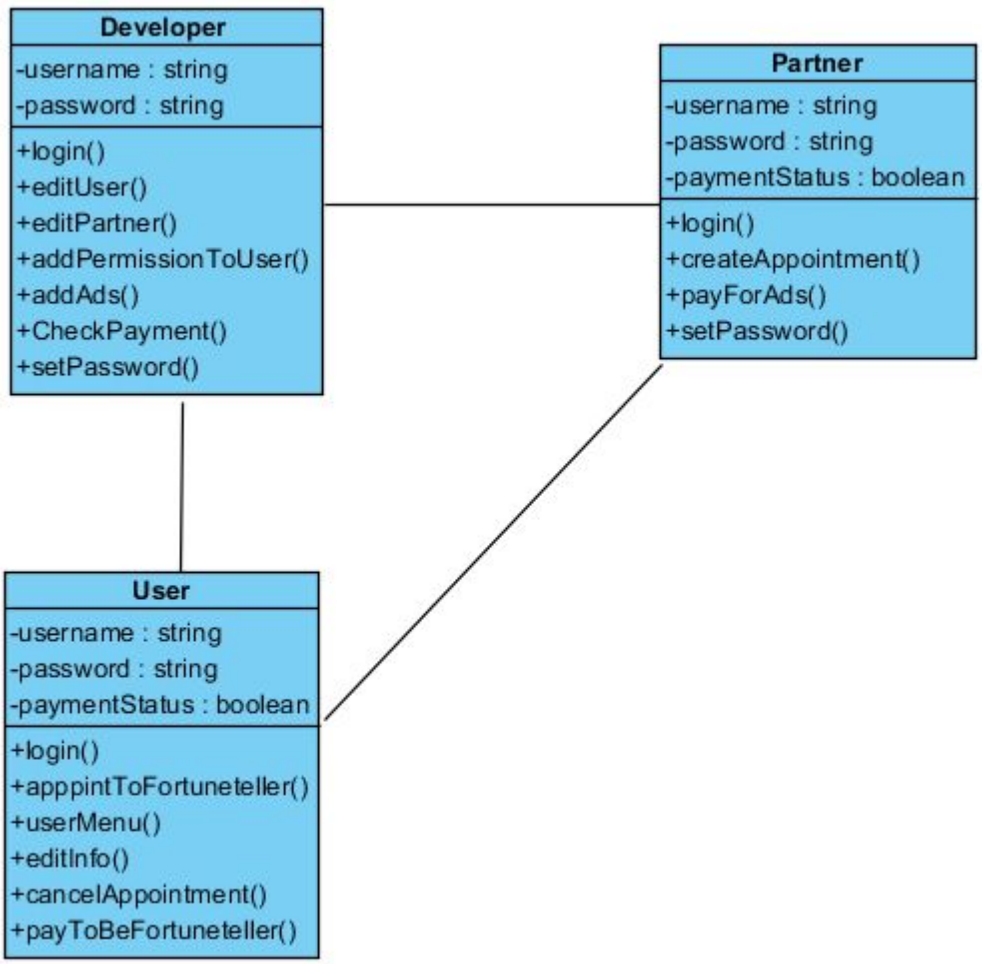
  USER

  - PROCESSING NARRATIVE (PSPEC) FOR COMPONENT

The component ‘user’, is our customer. They can use the application by free version or puchase version. There are 2 ways to access the application which are log in via Facebook or by guest. Users may share their results to thier own Facebook’s account in case of log in via Facebook. Futhermore, users can contact with real fortunet-eller in our application, set appointment and check their future.

  - COMPONENT 1 PROCESSING DETAIL

    - Design Class hierarchy for user



- Restrictions/limitations for component n

Users can only use application to foretell their future only. There are 2 ways to log in to an application. First users can log in via Facebook, another is by guest. Otherwise, sharing result on Facebook can be done by users who log in via Facebook.

- Performance issues for component n

The component have to verify that it was log in by a user.

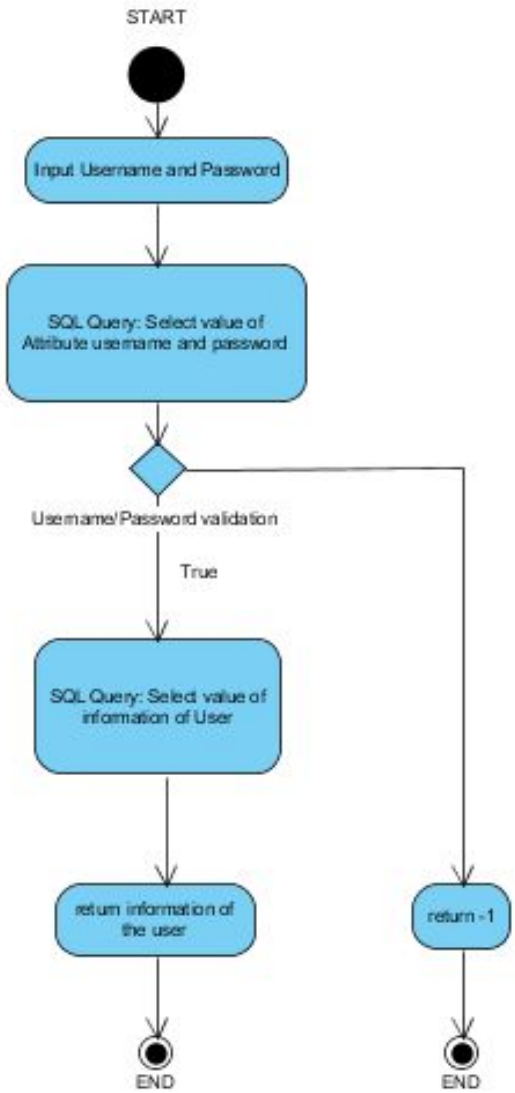
- Design constraints for component n

If the system is not loging in as a partner. It will show an error message for any case of mistaken for logging in and partner will not access a member site of appilcation.

- Processing detail for each operation of component n

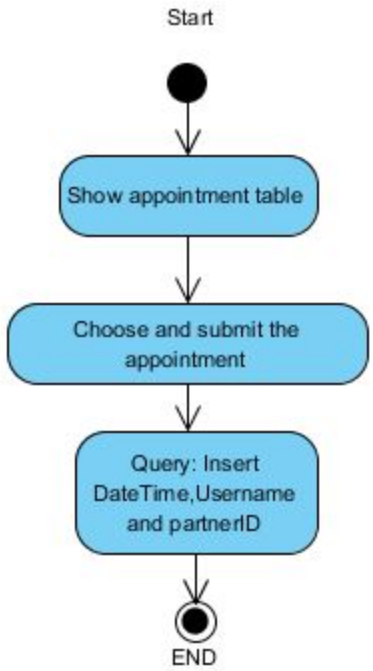
- login()

  This function is used to log in to an application by checking user's password in system. If there is matched password, system will return user's data. If not system will return -1.



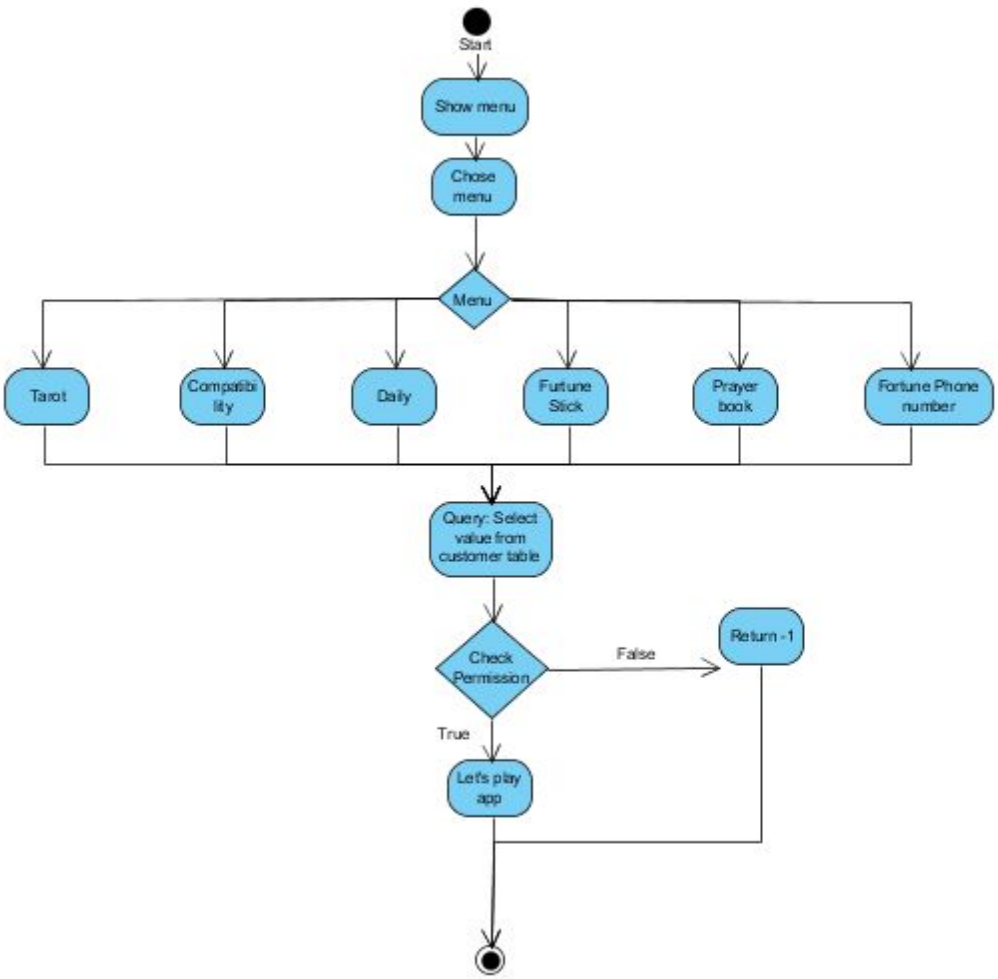
❏ appointToFortunteller() - (for user)

This function is use to appoint to fortuneteller.



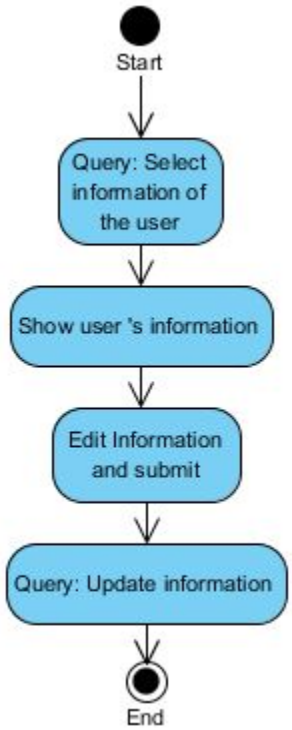
❏ userMenu() -(for user)

This function is used to select menu for link to play each application and check permission if the user have permission for that menu then go to play app,if not will return -1.



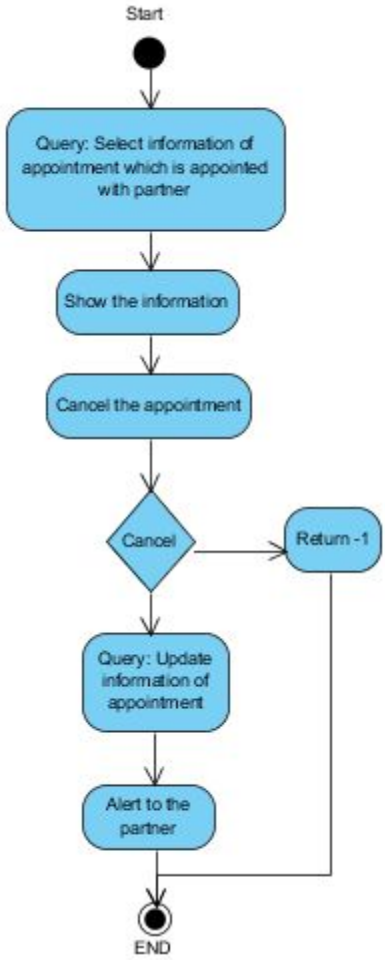
- editInfo() - (for user)

This function is used to edit information.



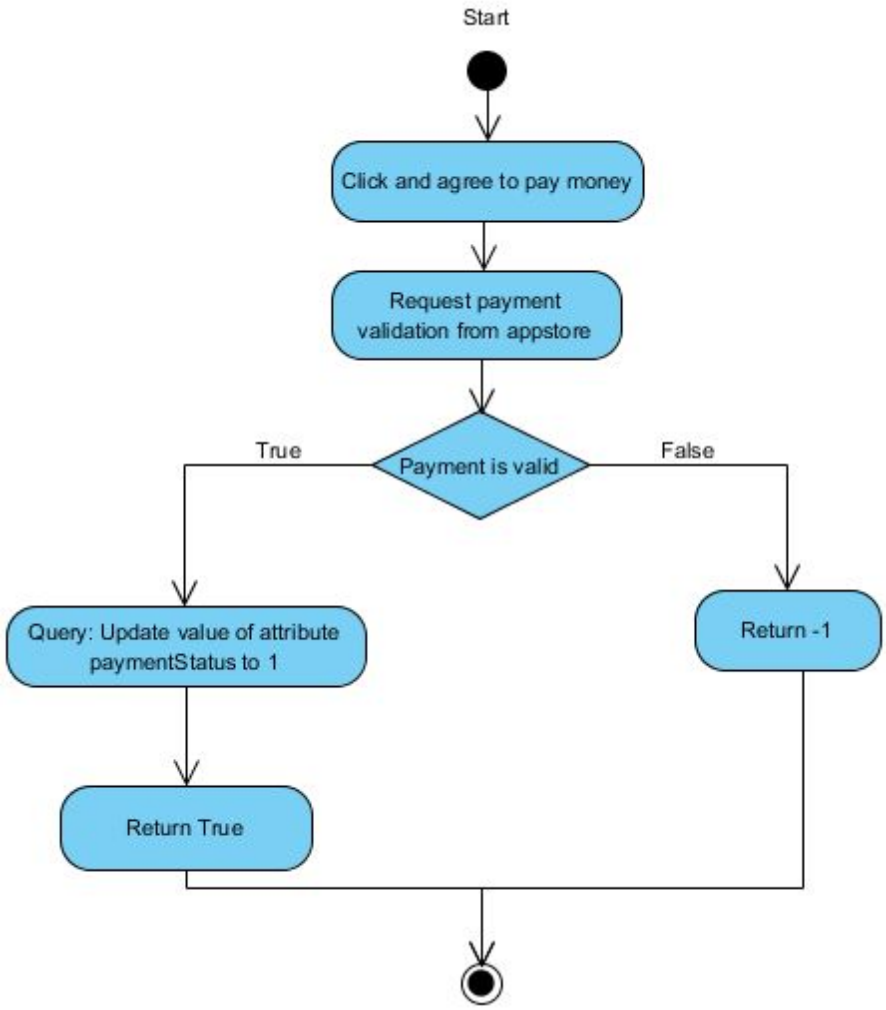
❏ cancelAppointment() - (for user)

This function is used to cancel the appointment.



- payToBeFortunteller - (For user)

This function is use to pay money to applicatio n to be fortuneteller.



- Processing narrative (PSPEC) for each operation

User

Process	Narrative
setPassword()	Use to set own password
userMenu()	Use to use application for each menu
viewPaymentResult()	Use to view payment information
cancelAppointmentTable()	Use to view appointment table
login()	Use to login to application and check authentication
payToBeFortunteller	Use to pay money to application to be fortuneteller.

- Algorithmic model (e.g., PDL) for each operation

For login
Start
Enter username and password
login to the app
End
For setPassword
Start
input old password
Input new password
Update information
Receive Result
End
For userMenu
Start
Choose menu
Chek permission
play application
End
For cancelAppointmentTable
Start
choose appointment
cancel appointment
update information of appointment
Receieve result
End

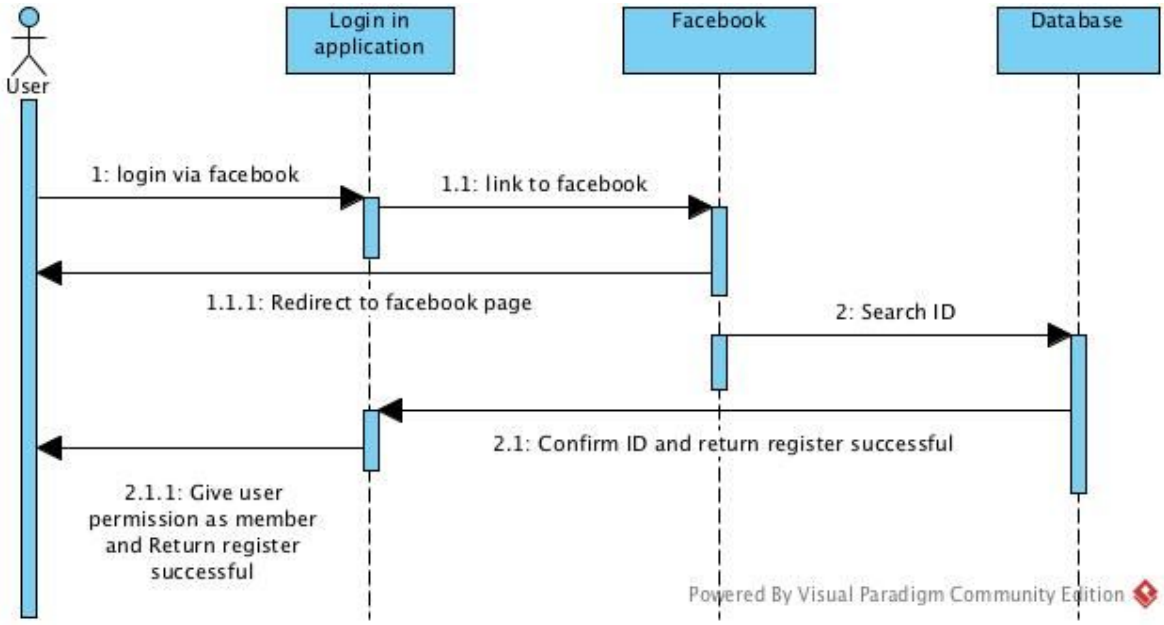
For payToBeFortuneteller

Start
Pay money to application
Check payment validation
Update payment status
Receive Result
End

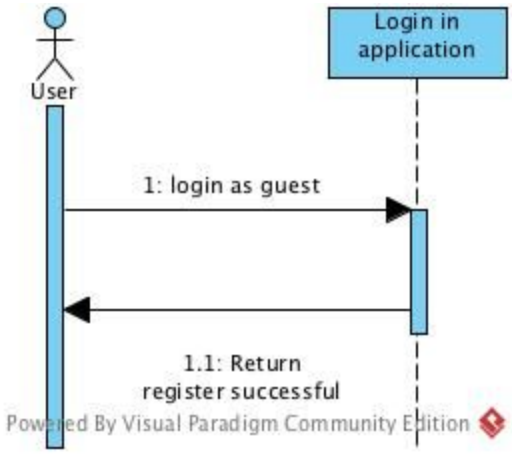
- COMPONENT 1 TEST POINTS LIST AND DESCRIPTION

Test	Description
Facebook login	ID and password of Facebook have to exist
Purchase unlock forecast function	Card number and money in Credit Card have to exist
Information in application	Don’t have any error or blank UI

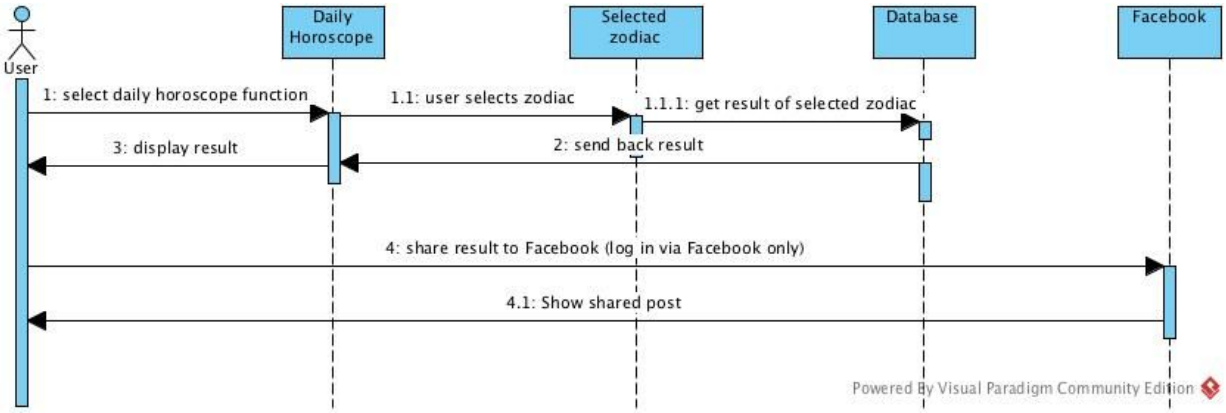
Log in via Facebook



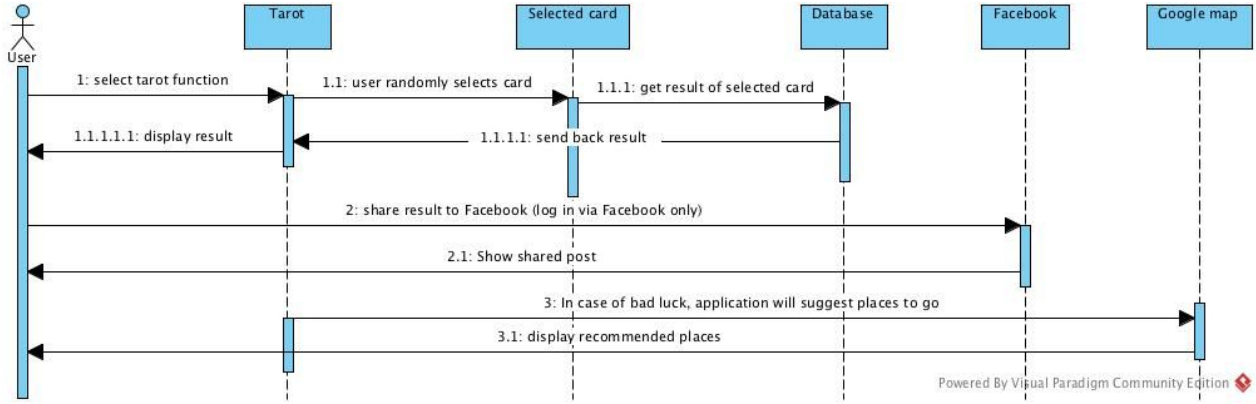
Log in as guest



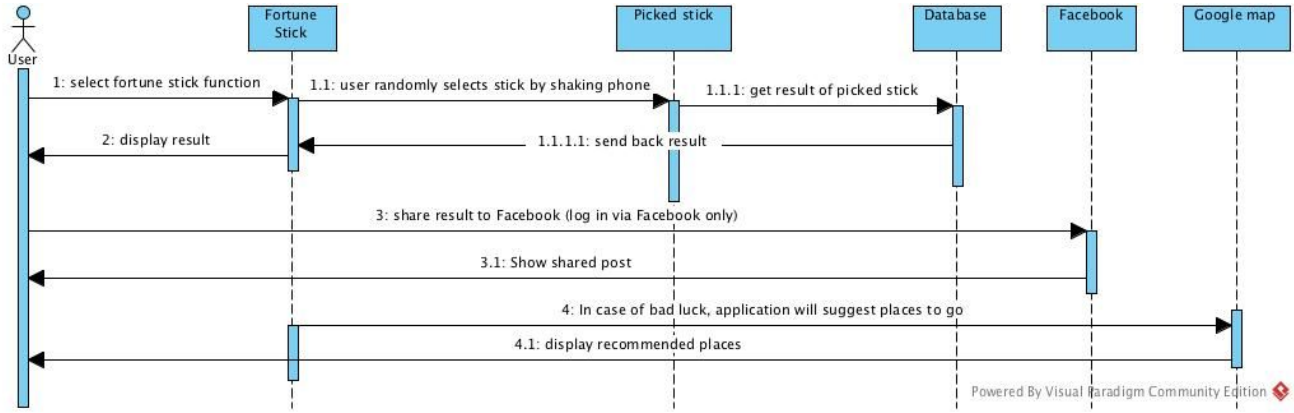
Daily Horoscope



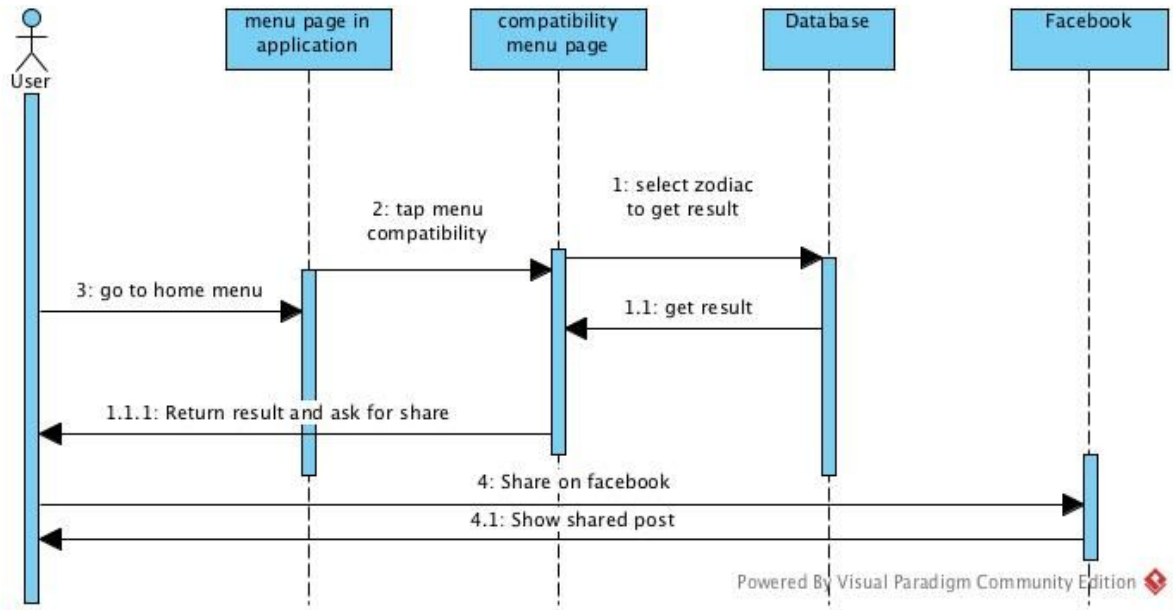
Tarot



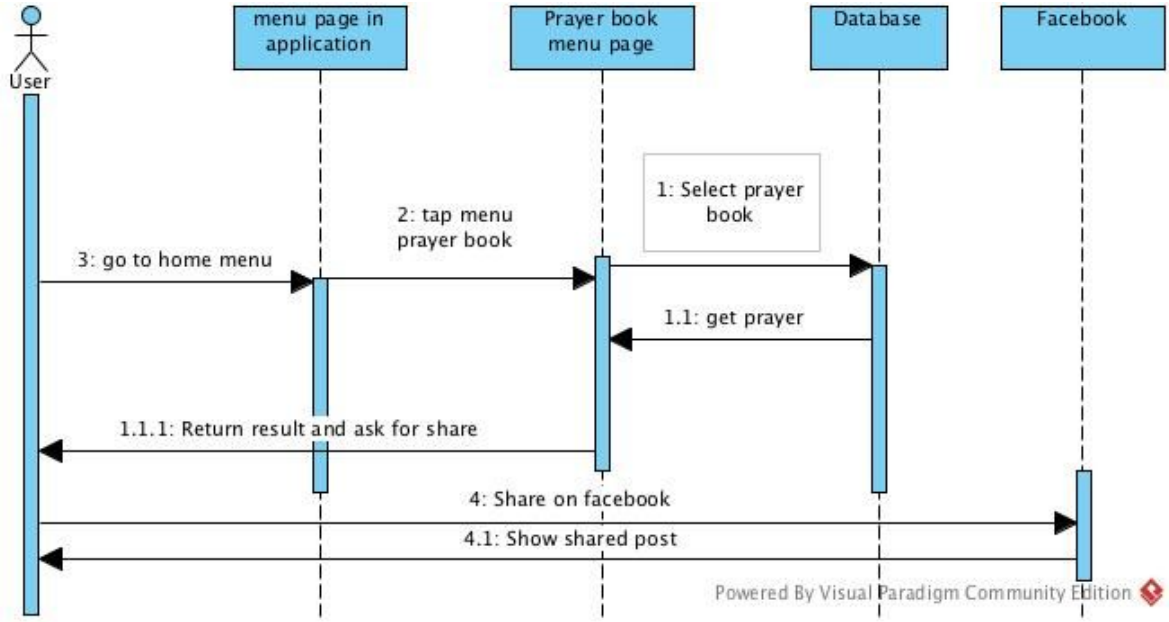
Fortune Stick



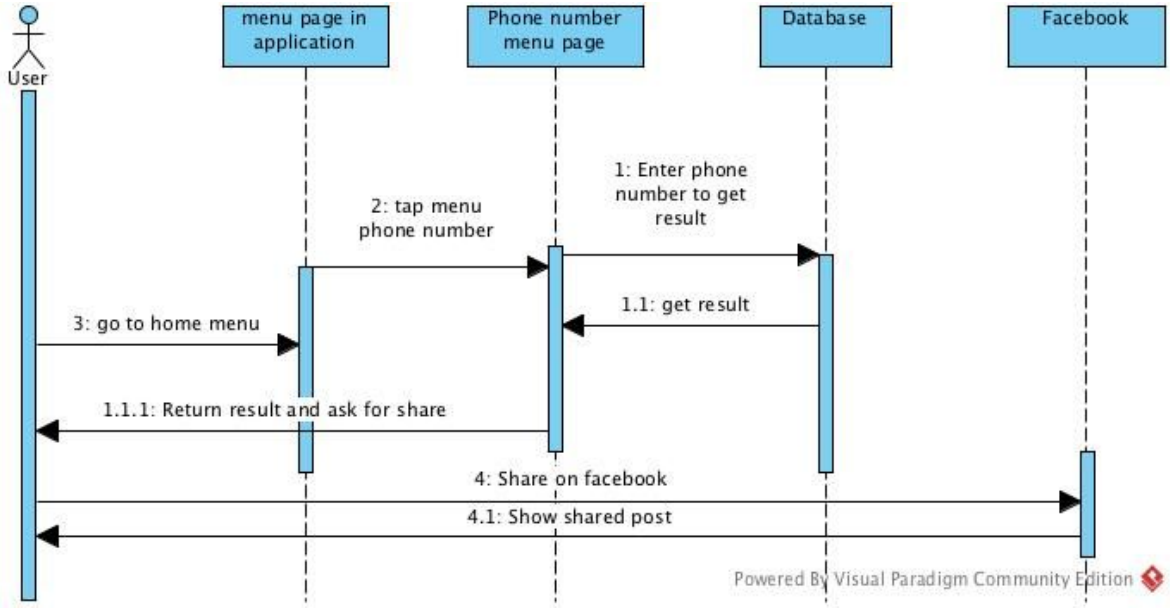
Compatibility



Prayer book

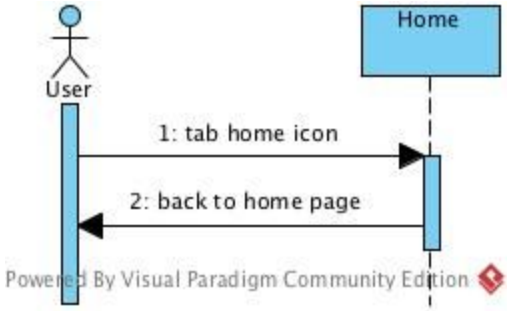


Phone number

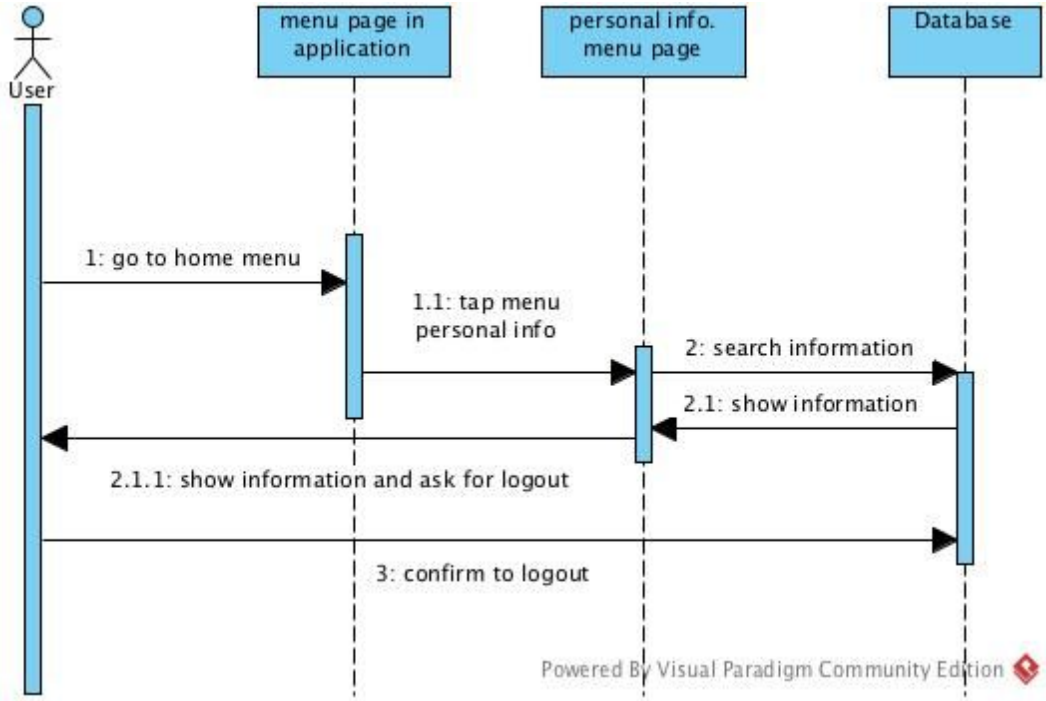


Menu tab

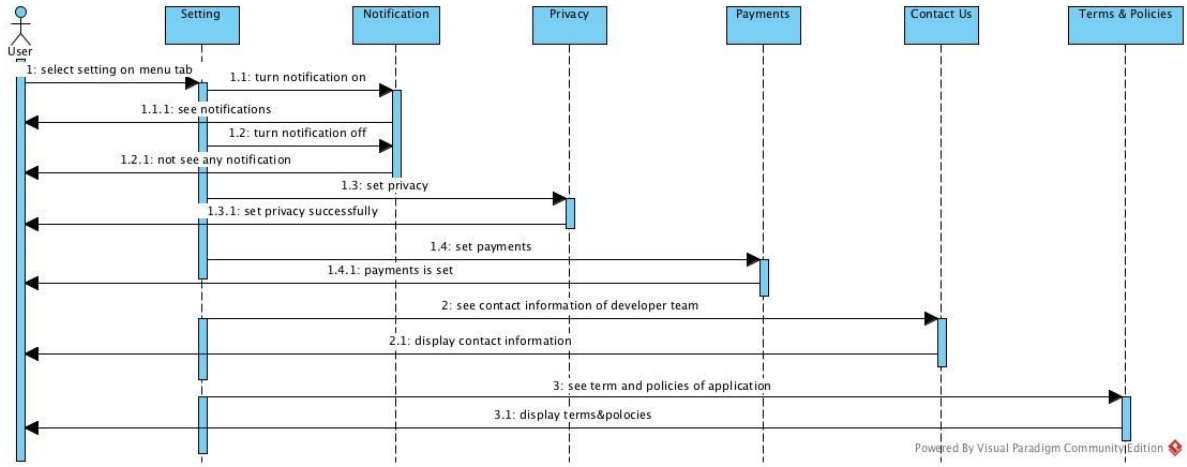
Home



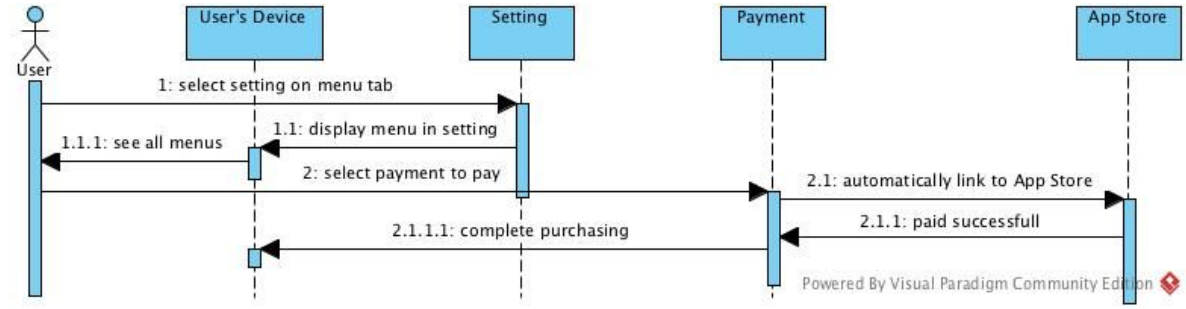
Personal information



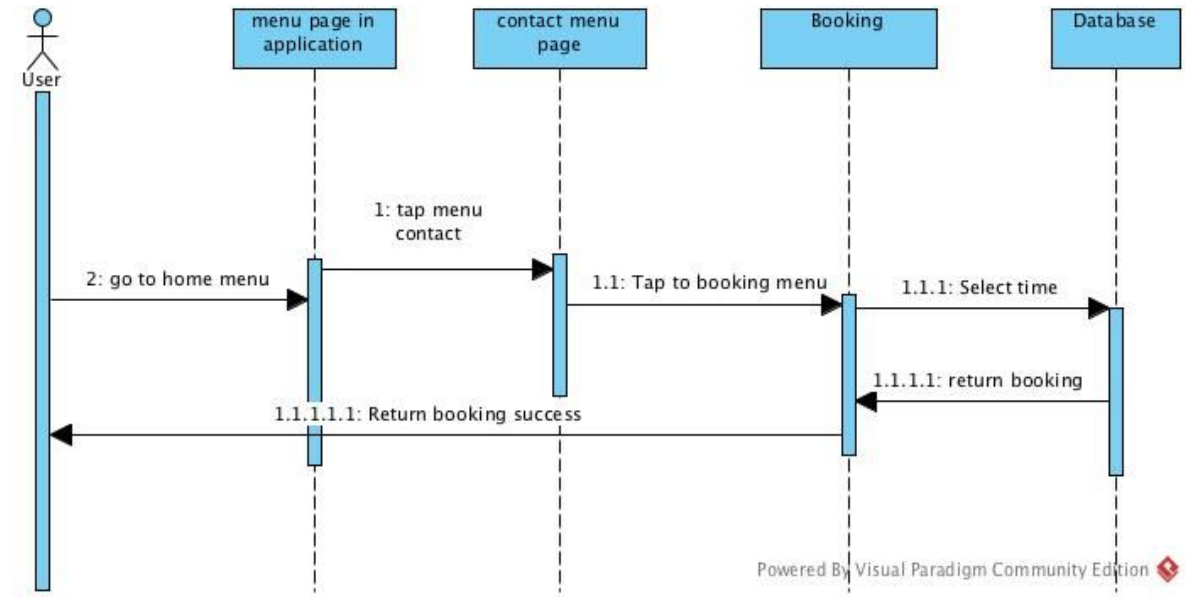
Setting



Setting : Payment



Contact



- COMPONENT 1 INTERFACE(S)

In this part, there are shown in Screen images and Description.

USER INTERFACE DESIGN *(design 2)*

- USER INTERFACE DESIGN RULES

User interface is one of a important part of our application, we decided to use Shneiderman's "Eight Golden Rules of Interface Design". These principle will help us to create a well designed user interface and also improve the usability of the system. It was seperated into 8 principles which are;

1. *Strive for consistency*
2. *Cater to universal usability*
3. *Offer informative feedback*
4. *Design dialog to yield closure*
5. *Offer simple error handling / Prevent errors*
6. *Permit easy reversal od actions*
7. *Support internal locus of control*
8. *Reduce short-term memory load*

1. Strive for consistency

Consistent sequences of actions should be required in similar situations; identical terminology should be used in prompts, menus, and help screens; and consistent commands should be employed throughout.

2. Cater to universal usability

As the frequency of use increases, so do the user's desires to reduce the number of interactions and to increase the pace of interaction. Abbreviations, function keys, hidden commands, and macro facilities are very helpful to an expert user.

3. Offer informative feedback

For every operator action, there should be some system feedback. For frequent and minor actions, the response can be modest, while for infrequent and major actions, the response should be more substantial.

4. Design dialog to yield closure

Sequences of actions should be organized into groups with a beginning, middle, and end. The informative feedback at the completion of a group of actions gives the operators the satisfaction of accomplishment, a sense of relief, the signal to drop contingency plans and options from their minds, and an indication that the way is clear to prepare for the next group of actions.

5. Offer simple error handling / Prevent errors

As much as possible, design the system so the user cannot make a serious error. If an error is made, the system should be able to detect the error and offer simple, comprehensible mechanisms for handling the error.

6. Permit easy reversal of actions

This feature relieves anxiety, since the user knows that errors can be undone; it thus encourages exploration of unfamiliar options. The units of reversibility may be a single action, a data entry, or a complete group of actions.

7. Support internal locus of control

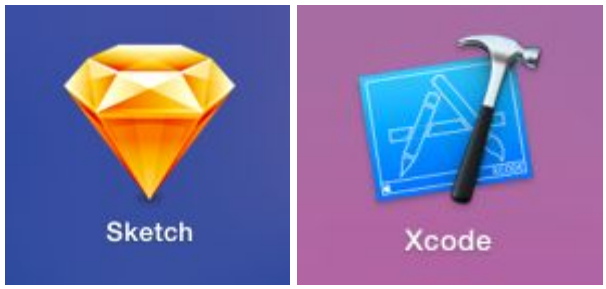
Experienced operators strongly desire the sense that they are in charge of the system and that the system responds to their actions. Design the system to make users the initiators of actions rather than the responders.

8. Reduce short-term memory load

The limitation of human information processing in short-term memory requires that displays be kept simple, multiple page displays be consolidated, window-motion frequency be reduced, and sufficient training time be allotted for codes, mnemonics, and sequences of actions.

- ## COMPONENTS AND DEVELOPMENT TOOLS USED

Sketch is used in a process to design and mock-up for user interface component. For coding, we used xCode to implement and develop our application. xCode is supported a Swift language that used for iOS application.



- SCREEN IMAGES AND DESCRIPTION

  Users

  - Home page

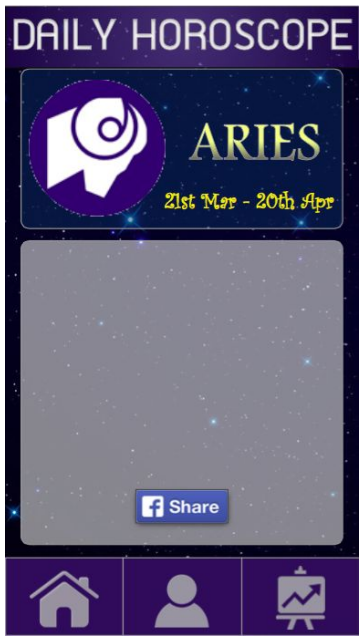
Users can log in to use our application by using their Facebook accounts, or log in as guest. That means users may not log in to use the application.

  - Menu

There 6 menus to select, those are Daily horoscope, Tarot cards, Compatibility, Fortune stick, Prayer book, and Phone number.

- Menu selecting; Daily horoscope (Home)

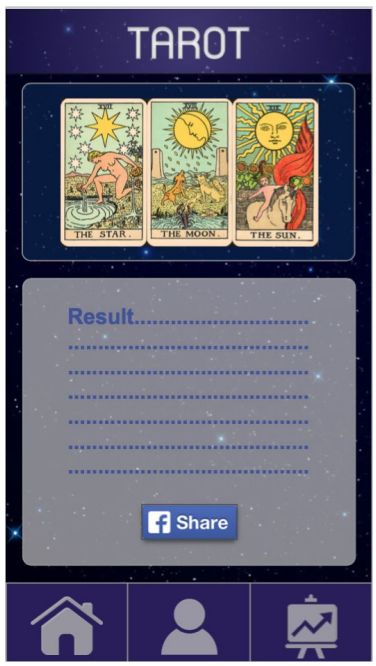
After selecting ‘Daily Horoscope’ at the menu, you will see a screen as a picture above. It shows zodiac, users may select their own zodiac to get a result.



For the result, the interface will look like this. At the bottom menu bar, a graph icon at the right side will show result in term of graph. And users can share result in facebook by tap on the share button.

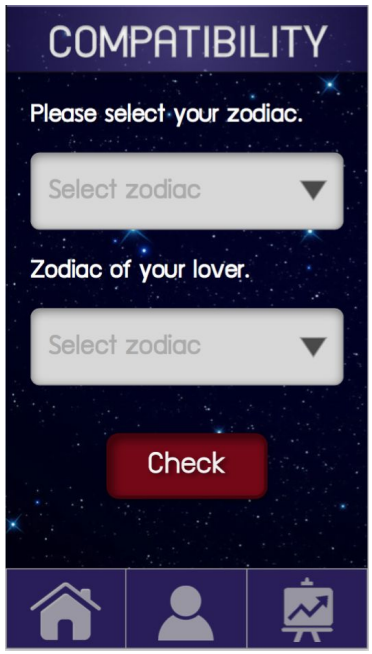
- Menu selecting; Tarot

After selecting ‘Tarot’’ at the menu, you will see a screen as a picture above. It shows cards, users must select 3 cards to get result.

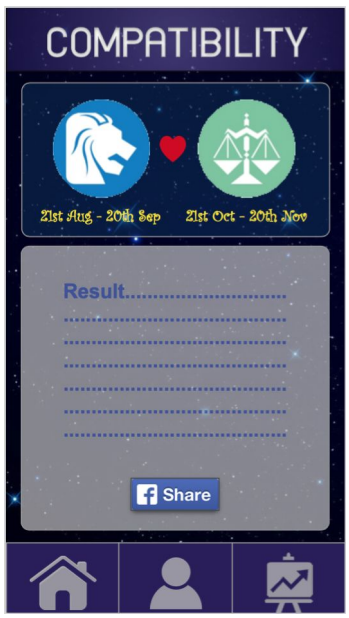


For the result, the interface will look like this. At the bottom menu bar, a graph icon at the right side will show result in term of graph. And users can share result in facebook by tap on the share button.

- Menu selecting; Compatibility



After selecting ‘Compatibility’ at the menu, you will see a screen as a picture above. It shows space for selecting zodiac, users must select their zodiac and their lover zodiac to get result.

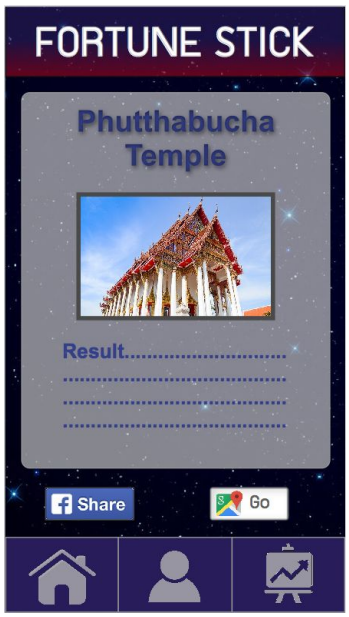


For the result, the interface will look like this. At the bottom menu bar, a graph icon at the right side will show result in term of graph. And users can share result in facebook by tap on the share button.

- Menu selecting; Fortune stick

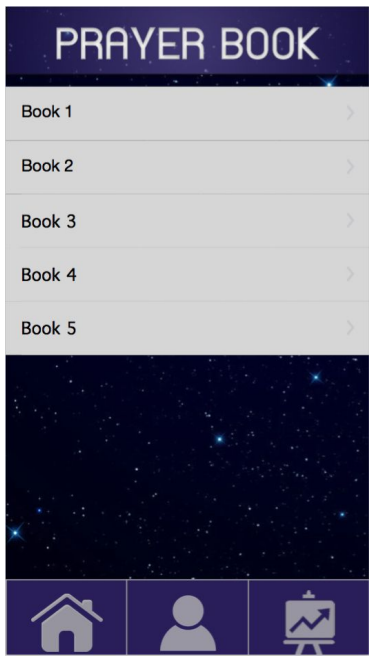
After selecting ‘Fortune stick’ at the menu, you will see a screen as a picture above.Users have to shake their phone to get the result.

For the result, the interface will look like this. At the bottom menu bar, a graph icon at the right side will show result in term of graph. Users can share result in facebook by tap on the share button. And users can tap exorcise for show how to remove unlucky with a ceremony.

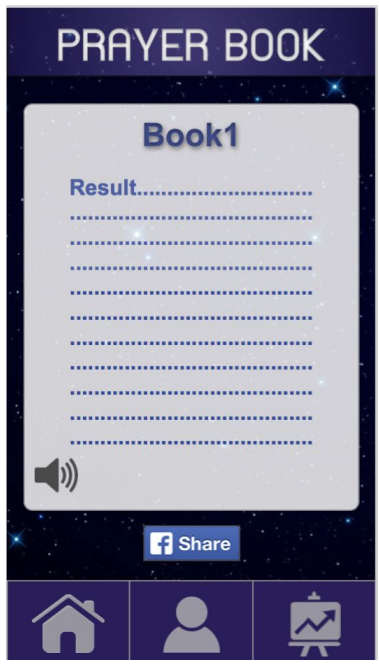


This is an exercise page that will show the temple for removing an bad luck by a ceremony and navigating to that place.

- Menu selecting; Prayer book

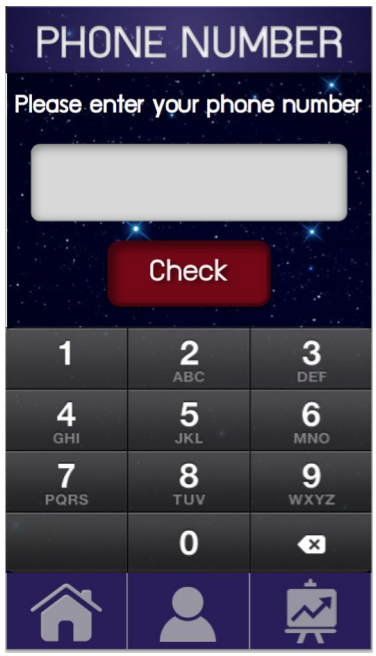


After selecting ‘Prayer Book’ at the menu, you will see a screen as a picture above. It shows prayer book menu, users must select the prayer book for get the prayer.

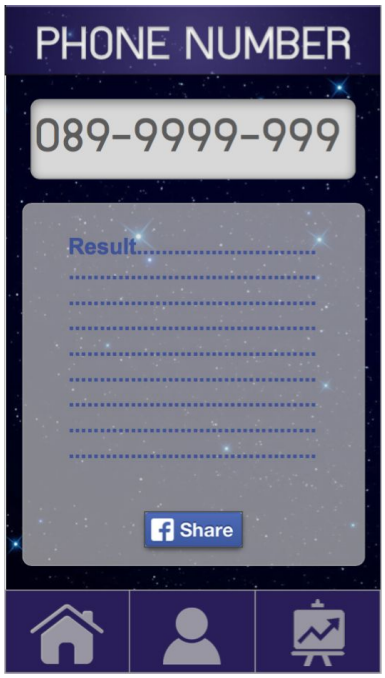


For the prayer, the interface will look like this. At the bottom menu bar, a graph icon at the right side will show result in term of graph. And users can share result in facebook by tap on the share button. Users can tap on audio icon for listen the prayer.

- Menu selecting; Phone number



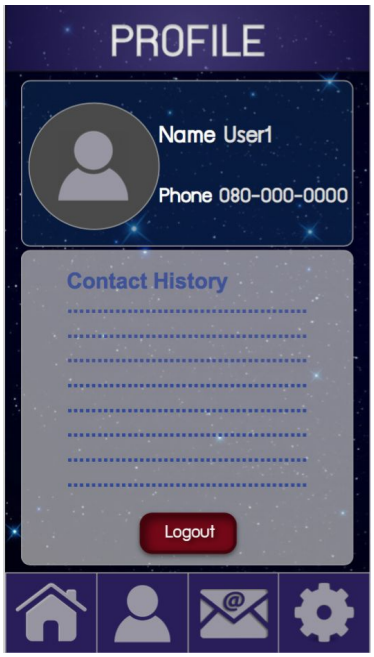
After selecting ‘Phone Number’ at the menu, you will see a screen as a picture above. It shows space for enter phone number, users must enter their phone number to get result.



For the result, the interface will look like this. At the bottom menu bar, a graph icon at the right side will show result in term of graph. And users can share result in facebook by tap on the share button.

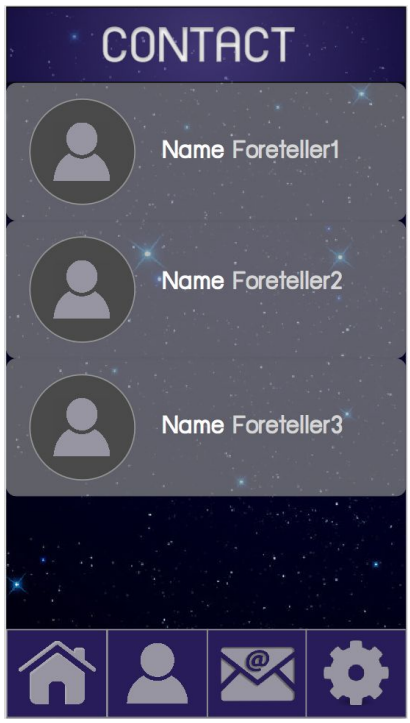
- Menu tab
    - Home

    - Personal information

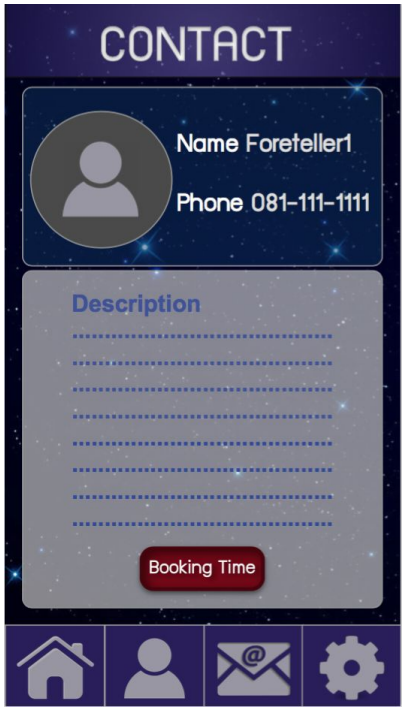


This page shows profile and history of users. User can logout by tapping on Logout button.

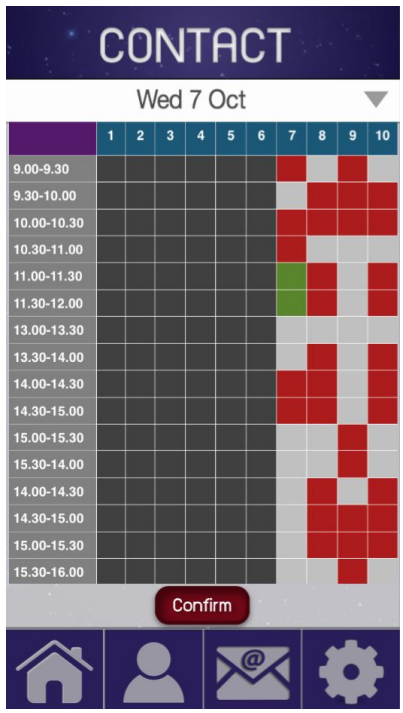
- Contact



This page shows list of foretellers. User can tap name of each foreteller for contact and booking time.

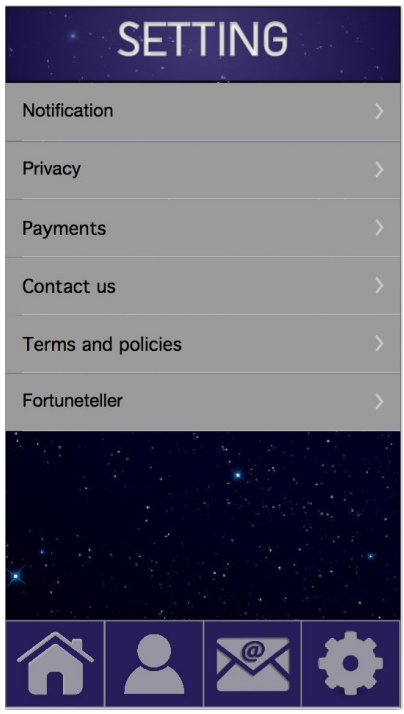


This page shows profile and description of foreteller. User can booking time to meet the foreteller by tapping on booking time button.



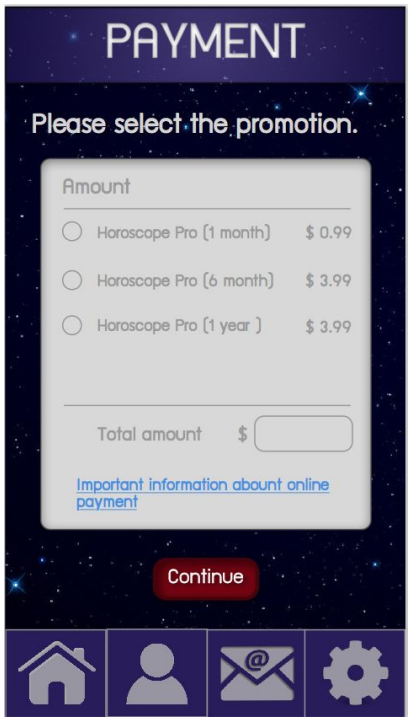
This page shows time table, users have to enter the time that they would like to meet the foreteller. Then, book the time and confirm request.

- Setting



This page shows setting menu. Users can set notification, privacy, payments, terms and policies and contact us.

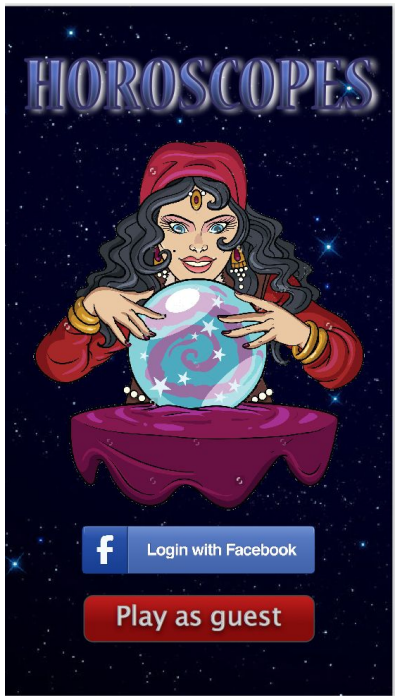
- Payment



This page show payment for each promotion. Users can select promotion and tap on Continue button. It will go to appstore for pay money.

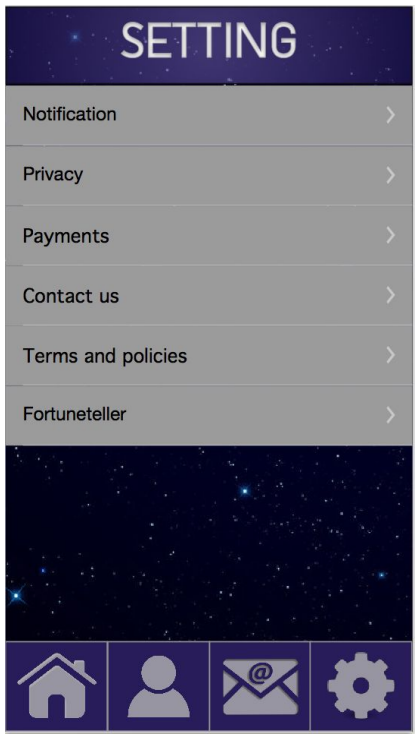
Partner(Fortuneteller)

- Login



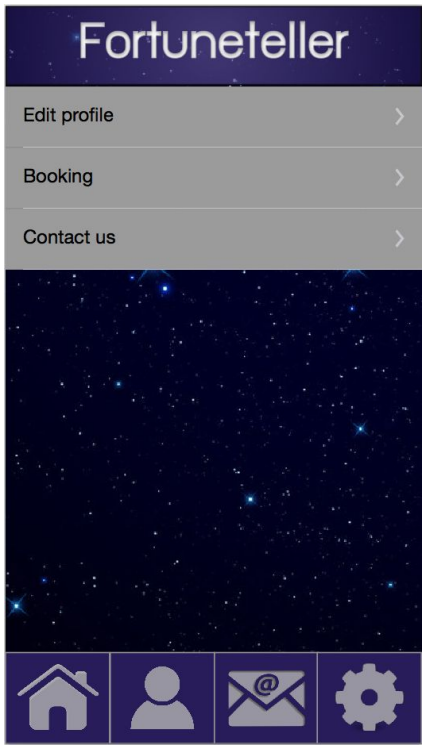
Fortuneteller must log in to use our application by using their Facebook accounts. After login fortuneteller can play functions in this application.

- Setting



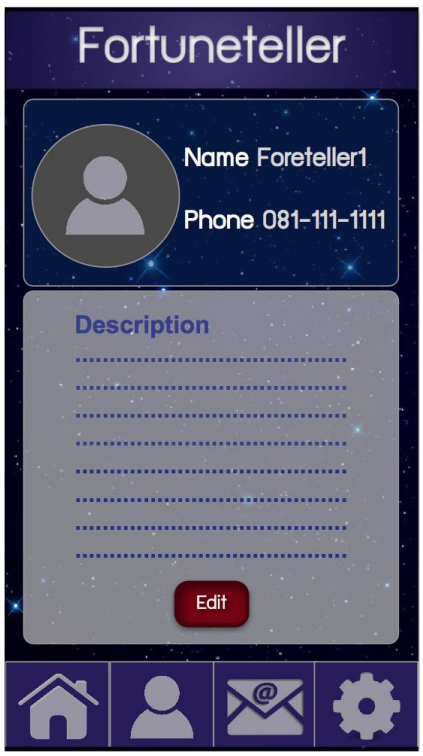
Fortuneteller should go to setting page and tap in ‘Fortuneteller’ for use the function that only for foreteller.ows main menu

- Main menu

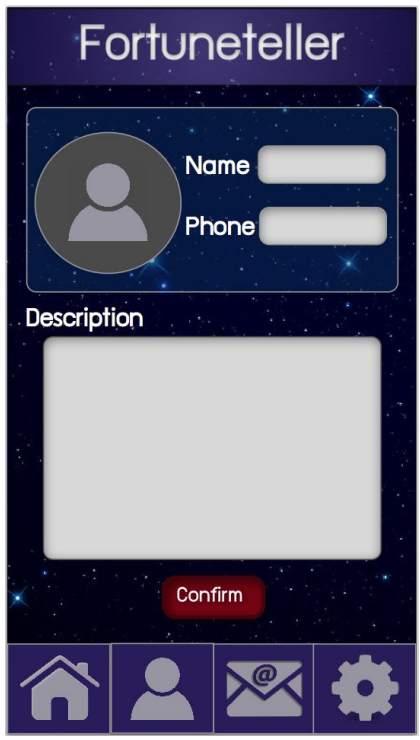


This page shows main menu for fortuneteller. They can edit their profile, check and edit booking time, and contact us for contact developer.

- Edit profile

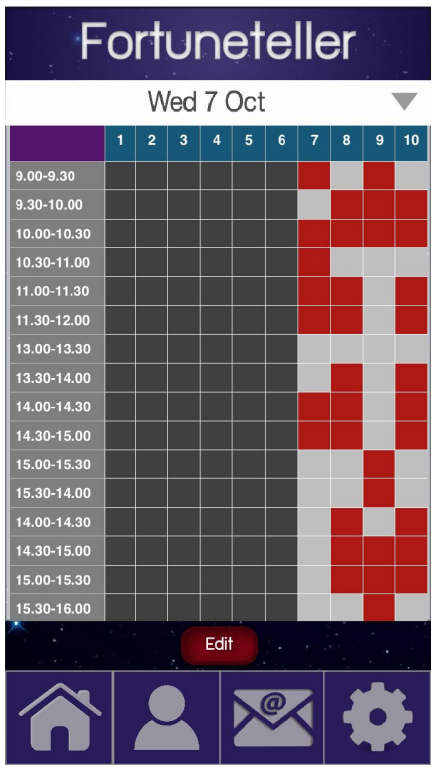


This page shows profile and description of foreteller. Fortuneteller can edit their profile by tapping Edit button.

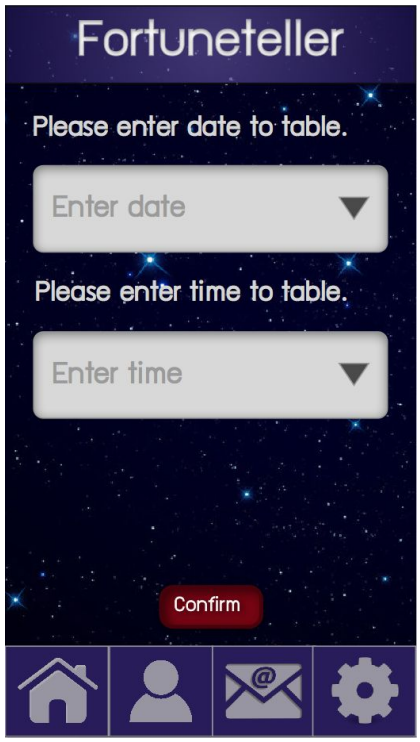


Fortuneteller can edit profile in this page and after edited foreteller must tap in Confirm button.

- Booking



This page shows booking time of user. That foreteller can know appointment and can edit the table by tapping Edit button.



After tap Edit button, foreteller can edit date and time for showing on table. When finished foreteller must tap on Confirm button.

OTHER INTERFACES DESIGN *(design 2)*

- HARDWARE INTERFACES DESIGN

  Hardware interface design is not necessary for the application.

- SOFTWARE INTERFACES DESIGN

  The application has to connect with Facebook, Google map and some payment API.

- COMMUNICATION INTERFACES DESIGN

  There is no communication interfaces design.

HOW TO USE FOR EACH SCENARIO *(design 2)*

- LOG IN

Actor : User
Goal : Log in to an application successfully

1. Selecting log in via Facebook or play as guest to get into an application.
2. Get to a menu page.

- DAIRY HOROSCOPE SCENARIO

Actor : User
Goal : Get today's fortune

1. Selecting zodiac.
2. After selected, user will get a today's fortune.
3. User may select a graph icon to generate reult into graph.
4. User can share their fortune on Facebook, in case of log in via Facebook only.
5. If user got a bad foretelling, application will provide a suggestion of recommended places via Google map to go pray.

- TAROT SCENARIO

Actor : User
Goal : Foretell user's life by tarot cards

1. Selecting 10 cards.
2. After selected, user will get a foretelling.
3. User may select a graph icon to generate reult into graph.
4. User can share their fortune on Facebook, in case of log in via Facebook only.
5. In case of bad luck, system will offer you recommended places via Google map to go pray.

- PRAYER BOOK SCENARIO

Actor : User
Goal : Access to prayer book successfully

1. After selecting prayer book at the menu, users can read all blessing chanting book that we provided.

- COMPATIBITY SCENARIO

Actor : User
Goal :

1. Users type in their names and their mates names.
2. System generates result from name, then display result of compatibility of two people.
3. User can share their fortune on Facebook, in case of log in via Facebook only.
4. Result may show in graph if users select graph icon.

- FORTUNE STICK SCENARIO

Actor : User
Goal : Get a foretelling of fortune stick

1. After getting into this page, users have to shake their phone to pull out a stick.
2. One stick have its number on it.
3. The system will show the number that users got, then it shows the foretelling.
4. Selecting graph icon, to get the result in term of graph.
5. If user got bad luck, system will suggest you places to go via Google map.
6. User can share their result to their own Facebook accounts, but they have to log in first.

- PHONE NUMBER SCENARIO

Actor : User
Goal : Get a foretelling result of their own telephone numbers

1. Type in their numbers
2. System generate result of each pair of numbers
3. User may share the result on Facebook, in case of log in via Facebook

- HOME SCENARIO

Actor : User
Goal : Back to home page successfully

1. Go to back home page after tab the home icon.

- PERSONAL INFORMATION SCENARIO

Actor : User and Developer
Goal : To keep user's data privately

1. In case of log in with Facebook, system will keep users information.
2. Developer can access into this part.
3. If uers request to contact with partners(fortune teller), only purchased users can do this activity.
4. Users can edit/delete their information.

- CONTACT SCENARIO

Actor : User and Partner
Goal : User can contact with fortune tellers

1. This scenario keeps purchased partners contact information.
2. Interested users can contact to partners(fortune tellers) directly to foretell their own luck.

- SETTING SCENARIO

Actor : User
Goal : Set privacy and general stuff in the application

1. Users can turn on/off notification from the application.
2. Users can change languages (Thai / English).

REQUIREMENTS VALIDATION MATRIX *(design 2)*

- FUNCTIONAL REQUIREMENTS CHECKLIST

Requirements	Purchased users	Free users	Developer	Partners
Log in	x	x		x

Daily Horoscope	x	x		
Tarot	x			
Fortune Stick	x			
Compatibility	x			
Prayer Book	x	x		
Phone Number	x			
Share result on Facebook	x	x		
Show prediction in graph	x	x		
Place suggestion on Google map	x	x		
Add, edit and delete functions			x	
Update version			x	
Provide partnet contact information			x	
Partner information	x	x	x	x

- NON-FUNCTIONAL REQUIREMENTS CHECKLIST

Requirements	Users	Developer	Partners
Apple devices (iPhone, iPad, and iPod touch)	x	x	x
iOS 7.1 and above	x	x	x
Mac OS X Yosemite		x	
xCode 7.0 or higher		x	

PROCESS MANUAL SPECIFICATIONS

- PROJECT PLAN AND MONITORING METHOD

We use Total Quality Management or TQM for monitoring our project. Our plan is to finished the project in time and meet all customer requirements. TQM is a comprehensive and structured approach to organizational management that seeks to improve the quality of products and services through ongoing refinements in response to continuous feedback. TQM processes are divided into four sequential categories: plan, do, check, and act (the PDCA cycle).

1.) Planning phase : people define the problem to be addressed, collect relevant data, and ascertain the problem's root cause.

2.) Doing phase : people develop and implement a solution, and decide upon a measurement to gauge its effectiveness.

3.) Checking phase : people confirm the results through before-and-after data comparison.

4.) Acting phase : people document their results, inform others about process changes, and make recommendations for the problem to be addressed in the next

PDCA cycle.

Week	Task	Delay
Jan 11-15, 2016	Brainstorming, Choose topic	-
Jan 18-22, 2016	Find out more information and planning, and finish docment of planning	-
Jan 25-29, 2016	Study and try to create diagrams via Visual Paradigm	-
Feb 1-5, 2016	Start a design(UI) of an application	-
Feb 8-12, 2016	Study about customer requirements	-
Feb 15-19, 2016	Editing the design and start creating platform	Did not create platforms
Feb 22-26, 2016	Review the customer requirements	-
Mar 1-4, 2016	Finish the final interface	-
Mar 7-11, 2016	Everyone starts a study about Swift, and start the software design document 1	Not everyone study Swift
Mar 14-18, 2016	Finish the software design document 1	-
Mar 21-25, 2016	Finish the software design document 2	Not finish
Mar 28, 2016 - Apr 1, 2016	Discuss about finished interface, review all works those are done, and double check on requirements	-
Apr 4-8, 2016	Start coding	Not start coding yet, still worked on software design document 2
Apr 11-15, 2016	Edit code and fix bugs	Long vacation

		(Songkran), still worked on software design document 2
Apr 18-22, 2016	Edit code and fix bugs	
Apr 25-29, 2016	Testing	
May 2-6, 2016	Edit code and fix bugs	
May 9-13, 2016	Testing	
May 16-20, 2016	Ready to present the application	

* note : blue weeks are plans.

Gantt Chart

Task	January				February				March				April				May			
	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4
Brainstorm, Choose topic																				
Planning																				
Creating diagram																				
Design UI																				
Study customer requirement																				
Editing design,create platform																				
Review customer requirement																				
Finish final interface																				
Study Swift																				
Software design document1																				
Start coding																				
Edit code and fixed bug 1																				
Testing1																				
Edit code and fixed bug 2																				
Testing2																				
Ready to present application																				

- EMPLOYEE WORK/TASK ASSIGNMENT PROCESS

Intuon	Pattarakitnitikul	CEO
Saran	Wongvisetsak	Head of Project
Supanun	Kaewsri	Designer
Aim	Manmalai	Developer
Varitta	Sombunying	Tester

- FINAL PROJECT COST METHOD WITH EXAMPLE

Direct cost

Software :

xCode (License) : Free for OS X
Sketch (License) : $99
Pages (License) : $19.99

Employee :

Cost per hour : 300 baht
Total hours a day : 6 hours
---
20 days per month : 36,000 baht / one employee

Example :

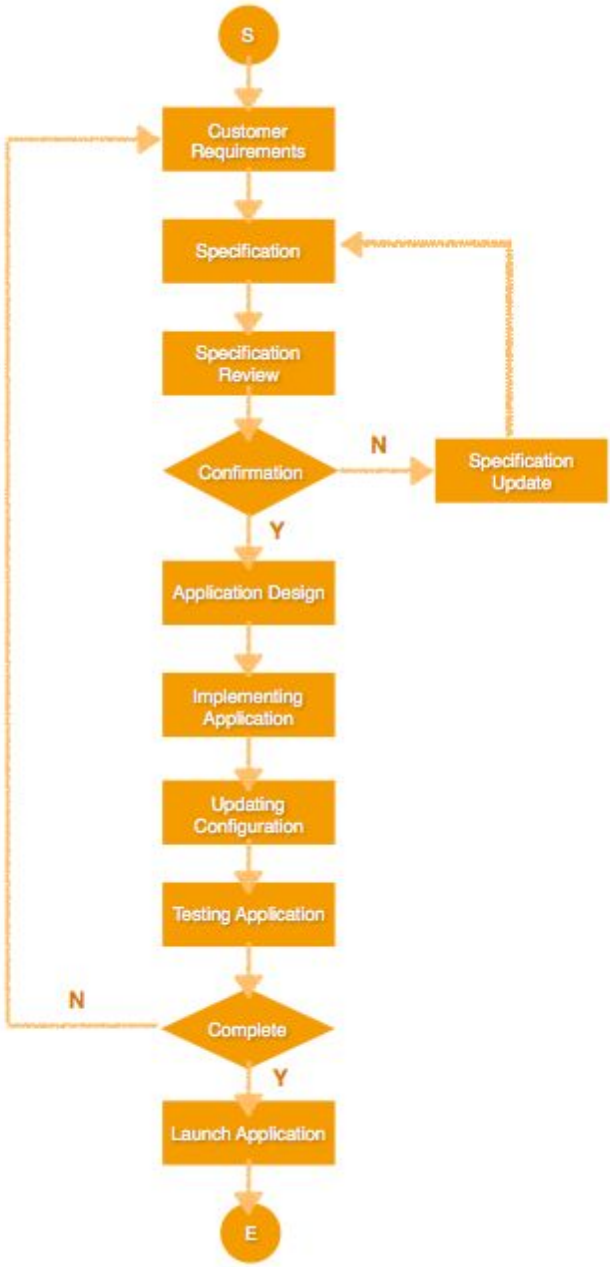
Time \ Date	Jan 11, 2016	Jan 12, 2016
10.00	Project Planning	Programming
11.00		
12.00		
13.00	Lunch	Lunch
14.00	Interface Design	Group Discussion
15.00		
16.00		

Indirect cost

Office rent : 15,000 baht / month
Meeting cost
Training cost : 3,000 baht / month

- REQUIREMENTS AND CHANGE MANAGEMENT PROCESS *(design 2)*

- CONFIGURATION MANAGEMENT PROCESS *(design 2)*



Description : Customer's requirements and needs are important in this part. In configuration management process, we used feedbacks of our customer to update the specification of application in case of it is not meet the requirement. Then, do a review to decide whether updated or work on a design. After designing, implementing process is started following by application testing, checking product to launch. If the application is ready, then lanunch it to community. If it is not ready, please go and check the customer's requirements and needs.

- MEASURES FOR SUCCESS IN TIMELY DELIVERY *(design 2)*

Process	Master Schedule			Actual Schedule		
	Duration (days)	Start (Date)	End (Date)	Duration (days)	Start (Date)	End (Date)
Start Meeting &Planning	2	18/01/2016	19/01/2016	1	19/01/2016	19/01/2016
Requirement gathering	2	18/01/2016	19/01/2016	2	18/01/2016	19/01/2016
Requirement document	2	18/01/2016	19/01/2016	2	18/01/2016	19/01/2016
Requirement document approval	4	20/01/2016	23/01/2016	10	20/01/2016	29/01/2016
Design UI	4	27/01/2016	31/01/2016	5	27/01/2016	31/01/2016
Design approval	5	27/01/2016	01/02/2016	12	27/01/2016	02/02/2016
Design document presentation	2	01/02/2016	02/02/201	2	01/02/201	02/02/201
Requirement analysis	6	03/02/2016	08/02/2016	6	03/02/2016	08/02/2016
Software design	2	08/02/2016	09/02/2016	2	08/02/2016	09/02/2016
Implementation on database	10	11/02/2016	20/02/2016	12	11/02/2016	22/02/2016
Implementation on server	14	08/02/2016	21/02/2016	14	08/02/2016	21/02/2016
User accptance testing	8	28/03/2016	04/04/2016	8	28/03/2016	04/04/2016
Traning and document	20	04/04/2016	23/04/2016	21	04/04/2016	24/04/2016
Product Run	6	25/04/2016	30/04/2016	7	27/04/2016	2/052016
Horoscope Application	87	18/01/2016	30/04/2016	104	19/01/2016	2/052016

Duration of master schedule = 87 days
Duration of actual schedule = 104 days
Calculate ; delay = (104-87)/87 = 0.1954 %

- USER ACCPETANCE PROCESS *(design 2)*
  - User acceptance Test Process
    The below diagram is show user acceptance's step process.

Description :

- Planning
  This step is the first thing we have concerned. To plan "HOROSCOPE APPLICATION", we had meeting and discussing about every details of this application.

- Designing Test Case
  Deriving test cases base on users (developers, partners, users)
  - ❏ Did guessing error?
  - ❏ Test exproratory

  Deriving test cases from requirement
  - ❏ Did use case correct?
  - ❏ Partition of equivalence
  - ❏ Diagram of state transition

  Deriving test cases from structure component
  - ❏ Path
  - ❏ Statement
  - ❏ Branch

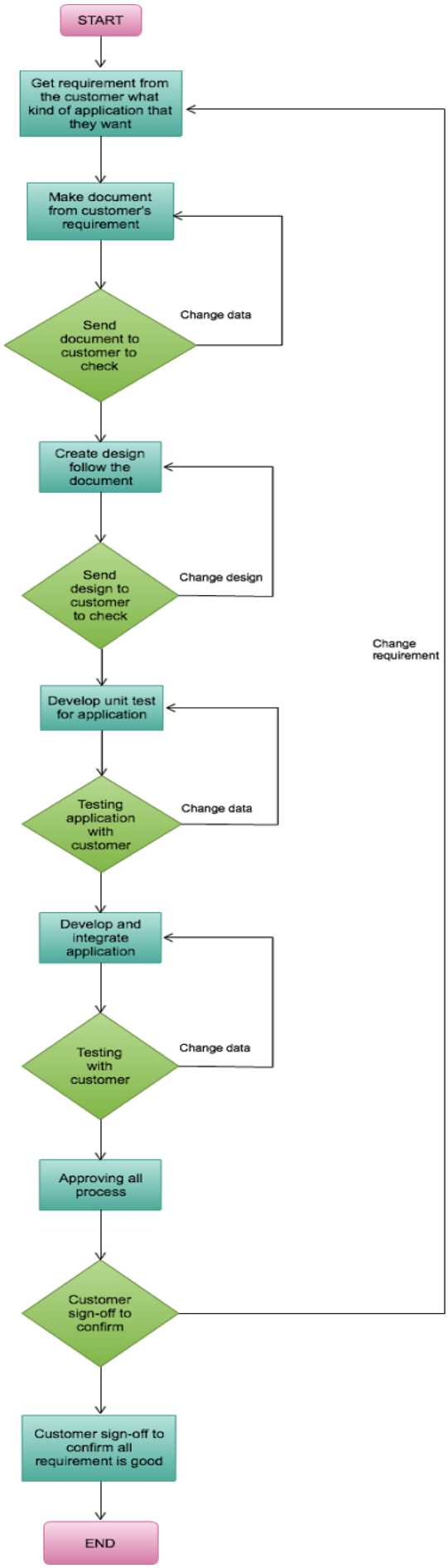
- Test Team Selection
  Test team of each department is they're suit that position or not.

- Test Case Execution and Documentation
  Check execution and documentation are they match with the planing.

- Bug Fixing
  If our application has a problem, we would adjust that problem by making this application work as we were plan or better.

- Sign-off

After everything was completed as we would like. Users who have acceptance test could be sign-off from the “HOROSCOPE APPLICATION” Project.


START
Get requirement from the customer what kind of application that they want
Make document from customer's requirement
Send document to customer to check
Change data
Create design follow the document
Send design to customer to check
Change design
Develop unit test for application
Testing application with customer
Change data
Develop and integrate application
Testing with customer
Change data
Approving all process
Customer sign-off to confirm
Change requirement
Customer sign-off to confirm all requirement is good
END

STEP 1 : *Get requirement from customer* -> Head of project has to list all requirements of customer
STEP 2 : *Make document from requirement* -> Make to check customer's requirement that is it correct data or not, and starting to make document
STEP 3 : *Send document to customer* -> Sending to customer because to make sure that is all of requirements are correct or would like to add or remove requirement. If the requirements are correct the customer has to sign-on to confirm .
STEP 4 : *Create design* -> Creating follow the document, this design has to guideline for developers to develop application
STEP 5 : *Send design to customer* -> The customer would evaluate the design document.
STEP 6 : *Develop unit test* -> Developer would be develop application by starting with a small for testing and check the reult before send to customer to accept.
STEP 7 : *Testing application with customer* -> Customer would have to accept or not. If customer don't accept, the developer have to solve the the testing.
STEP 8 : *Approving all processes* -> Customer will approve all processes, if application is like they would like.
STEP 9 : *Customer sign-off to approve* -> Customer has to approve all requirement before sign-off to confirm to done this project

IMPLEMENTATION

- Manual

On figure 1, selecting ‘Play as guest’ mode to enter the application without identify yourself. Then users will get into menu page to determine which function they would like to play on figure 2.

Figure 1

Figure 2

In this manual, Phone Number is selected. On figure 3, users have to enter their numbers to check whether they have good or bad numbers like figure 4.

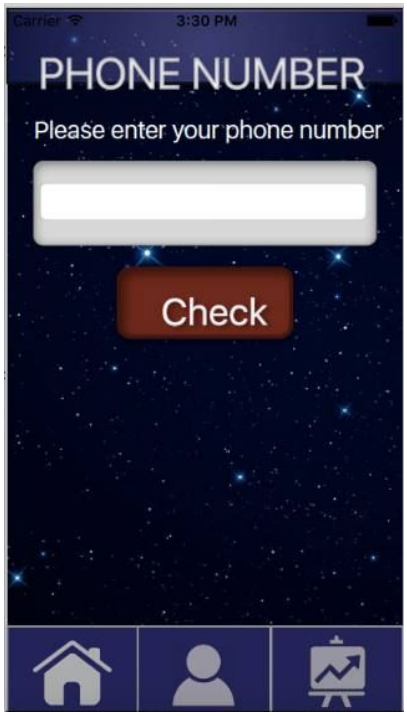


Figure 3

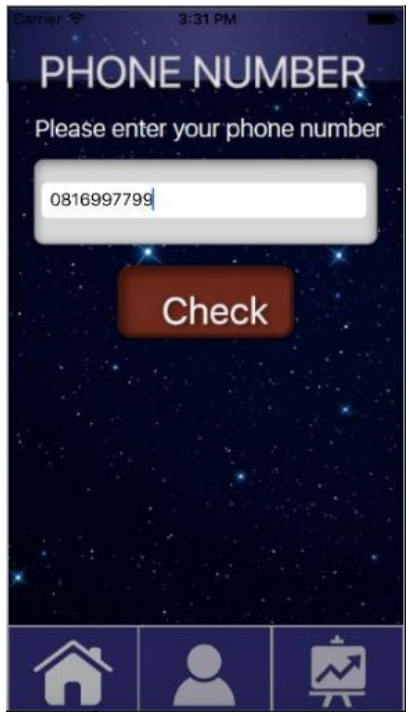


Figure 4

Another selected function is Daily Horoscope. Daily Horoscope forecasts a luck by zodiac. Users used to select their own zodiac to see a result as two figures below.

Figure 5

Figure 6

- Source Code

This capture is part of source code. Code at figure 7 shows a function to check how many digits of phone number, and figure 8 shows a piece of code of user interface.

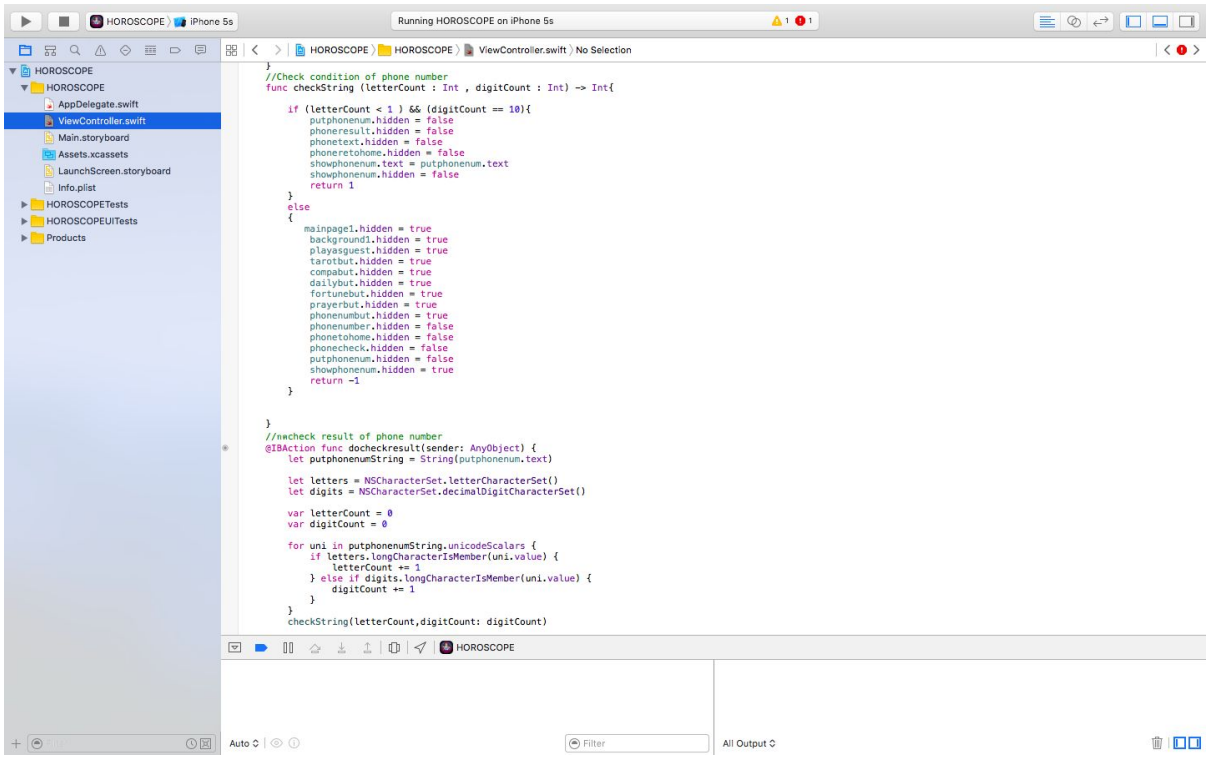


Figure 7

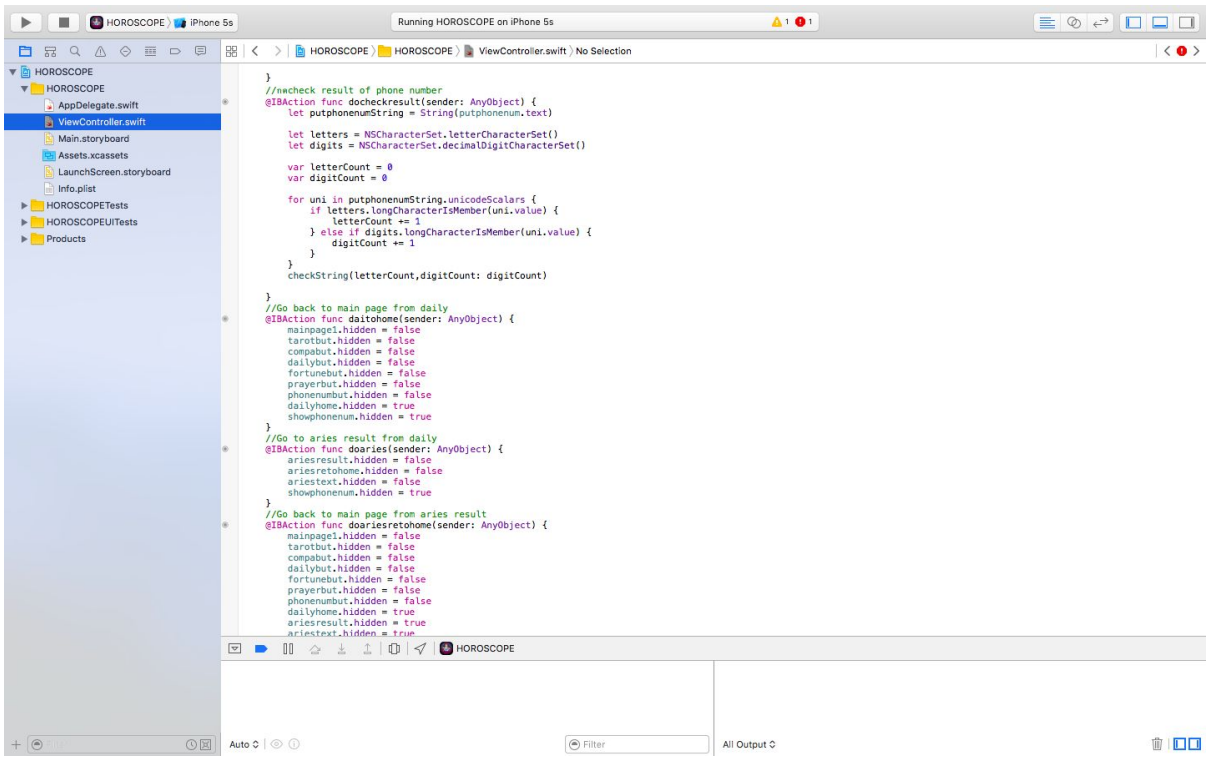
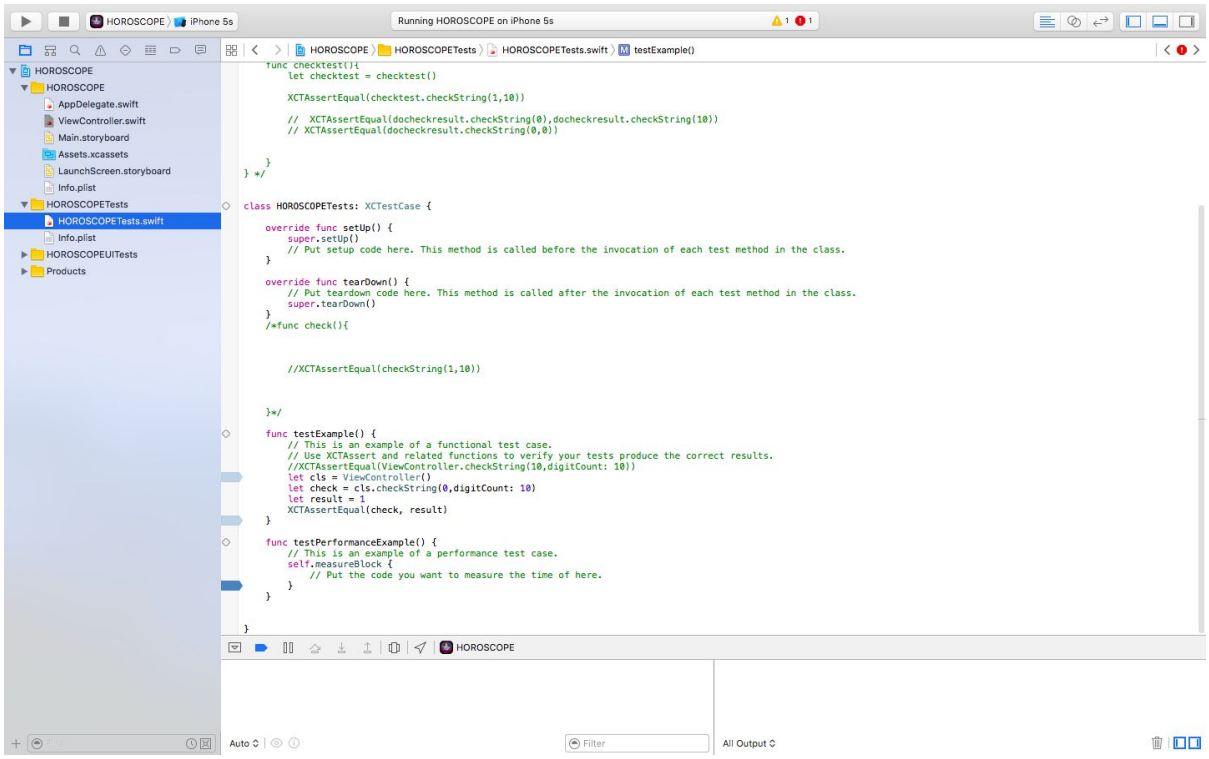


Figure 8

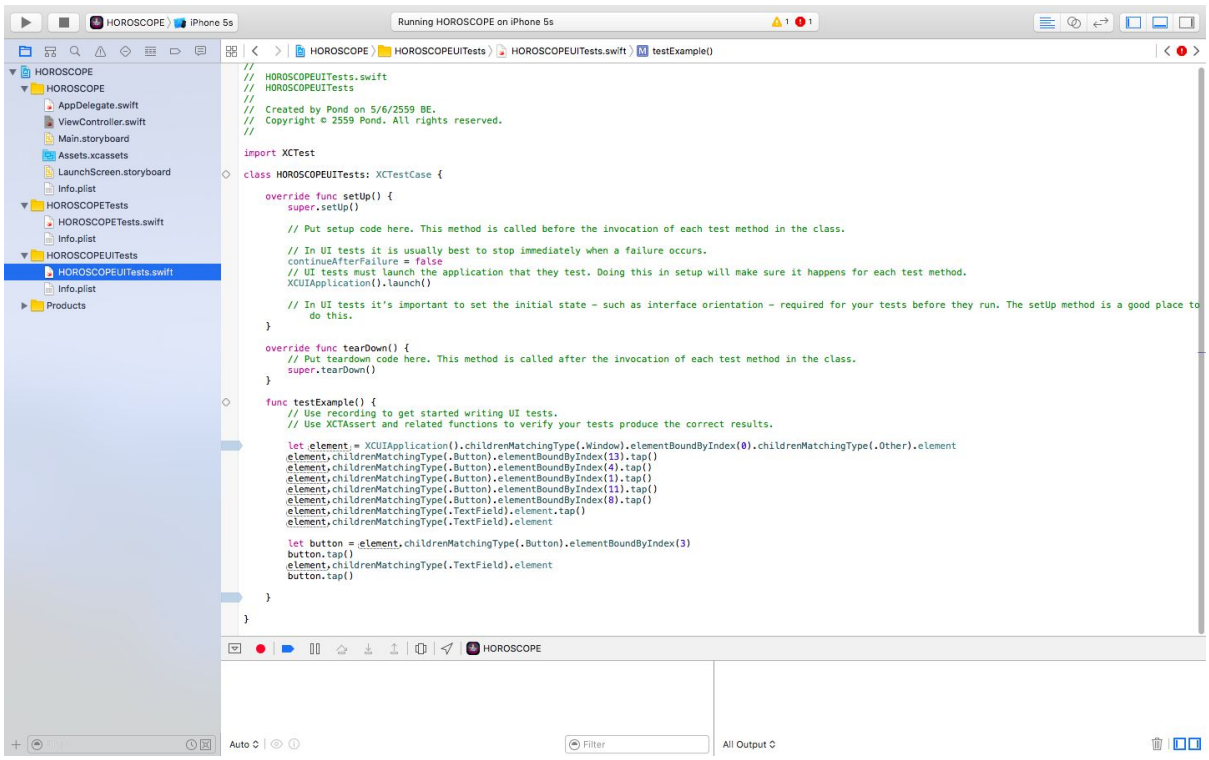
- Test Case

Test amount of phone number’s digit :



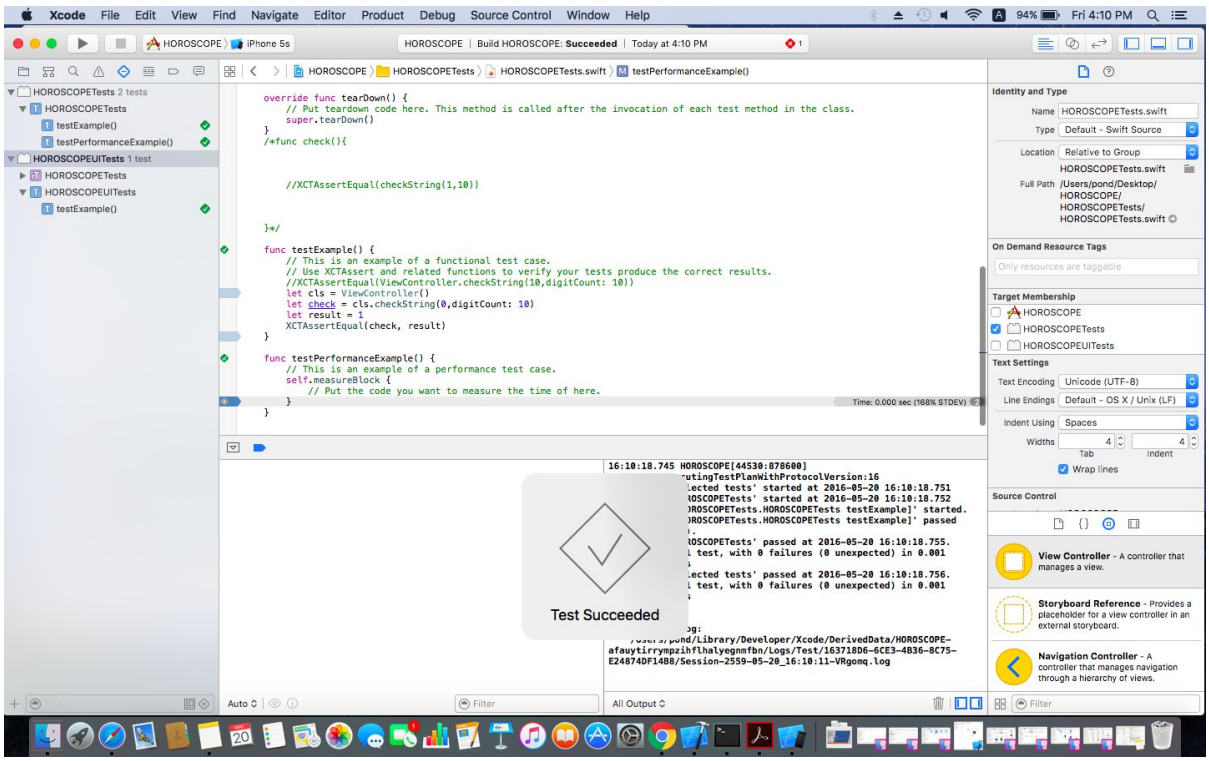
- Test UI

Test action of button that is work or not

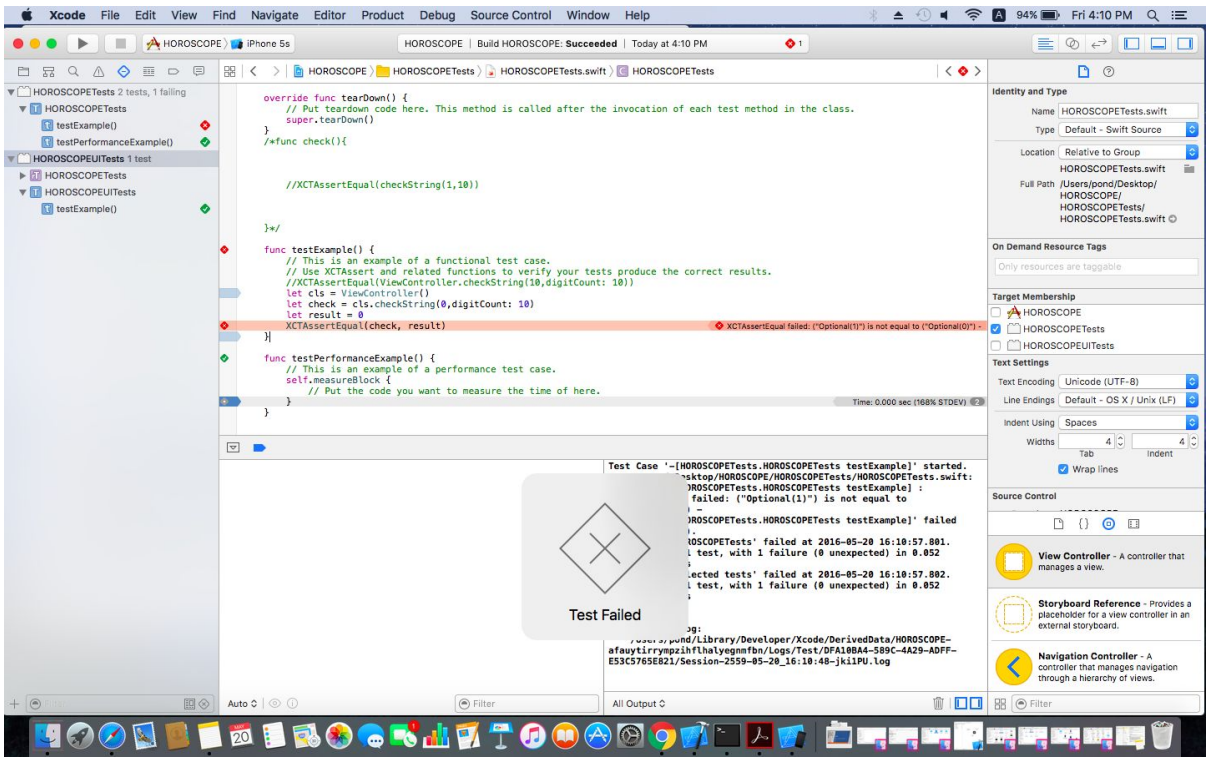


- Result of Testcase

This one is success from checking digit of phone number



This one is fail from checking digit of phone number



This one is success from action of each button