

Locust – Tracker for Safety

Software Architectural Design

Submitted to:
Asst. Prof. Ma. Rowena C. Solamo
Faculty Member
Department of Computer Science
College of Engineering
University of the Philippines, Diliman

Submitted by:
Celeste, Jimuel Jr.
Hernandez, Manolo
Puato, Ricardo

In partial fulfillment of Academic Requirements
for the course
CS 191 Software Engineering I
of the
1st Semester, AY 2019-2020



This work is licensed under a [Creative Commons Attribution-ShareAlike 4.0 International License](https://creativecommons.org/licenses/by-sa/4.0/).

Unique Reference:

The documents are stored in the GitHub repository (<https://github.com/manologhernandez/SOS/tree/master/03-Design%20Engineering>) referenced with filename *Locust - Architectural Design.pdf*.

Purpose:

To show the initial portion of the Revised Software Architecture. This includes the Software Architecture Model, as well as all the UI, Controller, DAO, Transfer, and Data Source Packages.

Audience:

Asst. Prof. Ma. Rowena C. Solamo

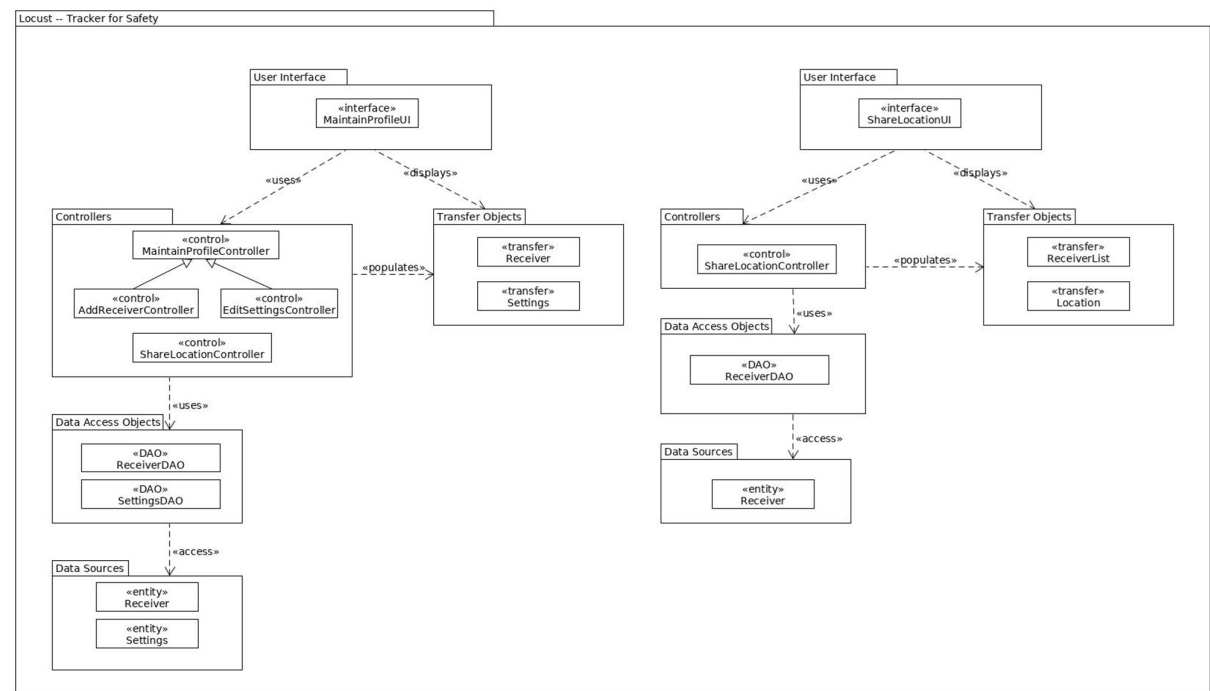
Revision Control:

| <i>Revision Date</i> | <i>Person Responsible</i> | <i>Version Number</i> | <i>Contribution/Modification</i> |
|-----------------------------|----------------------------------|------------------------------|--|
| 11/01/19 | Jimuel Celeste, Jr. | 1.0 | Initial Document; Added information on <i>User Interface Package, Transfer Objects Package, Controllers Package, Data Access Objects Packages, Data Sources Package</i> . |
| 11/01/19 | Manolo Hernandez | 2.0 | Validated attributes and responsibilities for objects in <i>User Interface Package, Transfer Objects Package, Controllers Package, Data Access Objects Packages, Data Sources Package</i> . Formatted the document. Added purpose and target audience. |

System Name: Locust – Tracker for Safety

Description: *Locust – Tracker for Safety* is a location sharing Android application. It would let the user to automate the process of location sharing. The application would get the user's current location via GPS then share the same via SMS. Of course, with the consent of the user; assuring that the full control is on the hands of the users.

Revised Software Architecture Model:



User Interface Package:

| Screen Name | Description |
|-------------------|--|
| MaintainProfileUI | <p>This is the interface of the user to the application whenever the user wants to maintain her/his profile.</p> <p><u>Attribute:</u></p> <pre>private TextField name; private Label nameLB; private TextField phone_number; private Label phone_numberLB;</pre> <p><u>Responsibilities:</u></p> <pre>public void enterReceiverData(String name, String phone_number); public void reviewReceiverData(); public void submitReceiverData(Receiver receiver); public void terminateProcess(); public void selectEditSettings(); public void makeModifications(); public bool reviewModifications(); public bool confirmModifications(Settings settings); public bool cancelModifications(Settings settings);</pre> |
| ShareLocationUI | <p>This is a interface shown to the user whenever he/she wants to share location.</p> <p><u>Attributes:</u></p> <pre>private Button shareLocationBT; private Button cancelShareLocationBT; private Button confirmShareLocationBT;</pre> <p><u>Responsibilities:</u></p> <pre>public void reviewReceiverList(); public bool confirmShareLocation(); public void cancelShareLocation();</pre> |

Controllers Package:

| Controller Name | Description |
|--|--|
| MaintainProfileController (abstract) | This is the control that maintain user's profile. It is considered as an abstract class. |
| AddReceiverController (extends MaintainProfileController) | <p>This is the control that adds a receiver to the application. It extends MaintainProfileController.</p> <p><u>Attributes:</u></p> <p>private Receiver receiver;</p> <p><u>Responsibilities:</u></p> <p>public void addReceiverData(Receiver receiver);</p> |
| EditSettingsController (extends MaintainProfileController) | <p>This is the control that edits the settings of the application. It extends MaintainProfileController.</p> <p><u>Attributes:</u></p> <p>private Settings settings;</p> <p><u>Responsibilities:</u></p> <p>public void terminateProcess();</p> <p>public void saveModifications(Settings settings);</p> |
| ShareLocationController | <p>This is the control that lets the user review the list of receivers and then shares the user's location.</p> <p><u>Attributes:</u></p> <p>private Location location;</p> <p>private ReceiverList receiverlist;</p> <p><u>Responsibilities:</u></p> <p>public Location getLocation();</p> <p>public void sendLocation(ReceiverList receiverlist);</p> <p>public void updateShareLocation(Location location);</p> <p>public void cancelShareLocation();</p> |

Data Access Objects Packages:

| DAO Name | Description |
|-------------|---|
| ReceiverDAO | <p>This data access object is responsible for getting receiver data from a file containing receivers' information named <i>Receiver.csv</i>.</p> <p><u>Attributes:</u></p> <p>private Receiver receiver;</p> <p><u>Methods:</u></p> <p>public void insertReceiver(Receiver receiver);</p> <p>public void updateReceiver(Receiver receiver);</p> <p>public void deleteReceiver(Receiver receiver);</p> |
| SettingsDAO | <p>This data access object is responsible for getting the current settings configuration data from the file named <i>Settings.csv</i>.</p> <p><u>Attributes:</u></p> <p>private Settings settings;</p> <p><u>Methods:</u></p> <p>public updateSettings(Settings settings);</p> |

Transfer Objects Package:

| Class Name | Description |
|--------------|---|
| Receiver | <p>This transfer object is responsible for holding the information of a receiver— receiver record.</p> <p><u>Attributes:</u></p> <p>private String name; private String phone_number;</p> <p><u>Methods:</u></p> <p>public void setReceiverName(String name); public void setReceiverPhoneNumber(String phone_number); public String getReceiverName(); public String getReceiverPhoneNumber();</p> |
| ReceiverList | <p>This transfer object is responsible for holding a list of receiver records.</p> <p><u>Attributes:</u></p> <p>private Receiver[] receivers;</p> <p><u>Methods:</u></p> <p>public void addReceiver(Receiver receiver); public Receiver[] getReceivers(); public void removeReceiver(Receiver receiver);</p> |
| Settings | <p>This transfer object is responsible for holding the current settings configuration data of the application.</p> <p><u>Attributes:</u></p> <p>private int saveLocationFrequency; private int noOfSavedLocations; private String defaultMessage;</p> <p><u>Methods:</u></p> <p>public void setSaveLocationFrequency(int saveLocationFrequency); public void setNoOfSavedLocations(int noOfSavedLocations); public void setDefaultMessage(String defaultMessage);</p> |
| Location | <p>This transfer object is responsible for holding the current location coordinates.</p> <p><u>Attributes:</u></p> <p>private float latitude; private float longitude;</p> <p><u>Responsibilities:</u></p> <p>public setLatitude(int latitude); public setLongitude(int longitude); public getLatitude(); public getLongitude();</p> |

Data Sources Package:

| File Name or Database Name | Description |
|----------------------------|---|
| Receiver.csv | <p>This is the file that contains the receiver records. It is structured as follows:</p> <ul style="list-style-type: none">• Each line represents a single record• Fields are separated by a comma (,)• Order of fields:<ol style="list-style-type: none">1. name2. phone_number |
| Settings.json | <p>This is the file that contains the current settings configuration of the application. It is structured as follows:</p> <ul style="list-style-type: none">• Every attribute is in a key : value pair• Attributes are grouped by braces ({,})• Attributes are separated by commas (,) |