Bright Light Systems Requirements v. 1.0

by Michael Gulenko & Dallas Winger

INTRODUCTION:

Bright Light Systems will be an app for Android OS and iOS that allows users to take full control over the Philips Hue light bulbs. Philips Hue is a system of wireless LED light bulbs that communicate with a Hue Bridge, a device that is connected to the user's Wi-Fi through the network's router. The application will allow users to set the color of the lights, manage color schemes, and setup light behavior for various events such as alarms, incoming text messages, emails, Facebook messages, location based events, and many more. The app will be targeting moderately sophisticated users, who have basic knowledge of how to use apps and smartphones.

Currently there are few complete options available on Google Play and iTunes markets. These apps give the user the ability to control the lights, but only to a certain extent, without implementing a comprehensive set of features. In addition, some of these apps were designed with an inferior user interface which results in a poor user experience. Also, in order to have light notifications when he/she receives messages on their phone, or perhaps to start a light show while listening favorite music, the user has to have at minimum 3 different apps running at the same time, which is extremely inconvenient.

Following is the competitive analysis of Phillips Hue App. The app evaluation has been done on HTC with the Android 4.0 phone.

Competitive analysis:

Phillips Hue App			
Platforms	Features	Widget	Price
· Android OS · IOS	· Color Control · Scenes · Alarms/Timer · Geofencing	Yes	Free

- Feature Analysis:

  Phillips Hue allows it's users to change the light bulb's color and brightness through a color palette and by picking a color from a picture. Colors and brightness that the user specified for the light bulbs can be stored as scenes. However, persistence of these settings occasionally gets corrupted, which causes incorrect color assignment to the light bulbs. Geofencing and timer features that turn the lights on/off based on the user location or a timer works properly. The alarm, on another hand, does not work while using the app due to the failing store alarm's settings. The user can setup a widget for Android OS, but it's also limited to allowing users to pick only 4 scenes.

- Conclusion:

  Phillips Hue is an app that provides a basic feature set, free of monetary cost. However, being limited in functionality and with a poor user interface, it provides a less than satisfactory user experience.

Bright Light Systems will resolve the issues described above by providing users with an intuitive user interface design, helpful widgets, and a comprehensive set of features which are all controlled from one place. This system will be developed and released as a commercial product. The cost of the application will be determined at a later date.

FEATURES AND CONCEPTS OVERVIEW:

To maintain understandability, key terms needed to help convey important concepts are provided below:

- **Hue Bridge** - a device that is directly connected to the user's router, which communicates any changes requested by the user to their Hue light bulbs.

- **Hue Bulb** – a single LED smart light bulb.

- ***Bright Light Trait*** - defines a Hue light bulb configuration: including color, density (saturation), and brightness.

- ***Bright Light Trigger*** - defines the specific conditions that need to be met in order to cause the **Event**.

- ***Bright Light Event*** - defines a change of a single light or group of lights; caused either directly by the user or by a user-defined **Trigger**

- ***Bright Light Groups*** - defines a group of Hue light bulbs. This will be used to apply **Event**(s) to multiple Hue light bulbs.

The main goal of the Bright Light Systems is to deliver an intuitive and easy way to setup and maintain Phillips Hue light bulb systems. This will be accomplished by implementing so called Bright Light Event(s); a concept that describes both a Bright Light Trait, which includes a color schema, and Bright Light Triggers which define the conditions to be met to initiate the **Event**. Behavior of each Hue light bulb can be set individually or as a user-defined set of Hue light bulbs.

Bright Light Events:

One of the key concepts of this app is the **Bright Light Event**. In order to create such events the user will have to specify two things: a **Trait** and a **Trigger**. A trait describes light characteristics such as color, brightness, and density of the light bulb or set of bulbs. A trigger, on the other hand, describes a condition(s) upon which a certain event will be fired off. A trait can be chosen from color themes previously specified by the user, or created on the fly; in which case upon completion of defining the Trait, the app will give an option for the user to save the created trait as a color scheme. Triggers can be selected from following: **Location Based Trigger**, **Message Notification Trigger**, and **Scheduled Trigger.**

Bright Light Triggers:

- Location Based Trigger:

  This trigger will be chosen by the user in order to create an event based on his/her location. The trigger determines which lights will be turned on/off based on if the user leaves or approaches his/her location. The user will be able to specify if the lights have to stay off when the user is back if the current time is before sunset. This will prevent the lights from turning back on if it's still light outside, but on other hand it will give users the option to specify if the lights should go back on if needed.

- Message Notification Trigger:

  This trigger will be responsible for notifying the user about a new message that arrived on his/her Facebook profile(s), specified email(s), or SMS/MMS on the device. The user also will be able to set a light behavior that will respond to this event such as blinking, changing colors.

- Scheduled Notification Trigger:

  This trigger will be responsible for notifying the user when the timer defined for this event has expired or when the system time, specified by the user, has been reached. The light behavior that will respond to this event will be blinking, changing colors, fading, and lights on/off.

Bright Light Systems Features:

- Color/Brightness Change
- Color/Brightness Transitions Over Time
- Theme Organizer
- Scheduling (Alarm or Timer based)
- Location Based Actions (Auto On/Off)
- Message Notification (Text, Email, Facebook, etc)
- Music Mode
- Homescreen Widget Control

While starting the application for the first time, Bright Light Systems will prompt the user to pair a Phillips Hue Bridge; a device that is connected to user's Wi-Fi, via router, which communicates any changes requested by the user to the Hue light bulbs. The paired Hue Bridge will be saved by the app and can be updated or deleted at any time. The Bright Light Systems application will store as many as ten different Hue Bridges that the user can connect to; allowing (theoretically) for up to 500 individual Hue light bulbs to be controlled from the same app. The user will be able to choose to which bridge he/she wants to connect to by default out of the available (already configured) bridges on device. This "Default" / "Connect Automatically" feature will be an option for any paired Hue Bridge and can be changed later through the app's settings.

Connecting to the Hue Bridge allows the user to be able to manage all Hue light bulbs that are controlled by that Hue Bridge. The user will be able to assign names to different Hue light bulbs to distinguish them from each other, such as "Living Room Ceiling" or "Floor Lamp", for example. The user may create different groups of Hue light bulbs, referred to as a Bright Light Group. Groups can also be named. The amount of Hue light bulbs in each Group will be between 1 and the maximum supported amount (up to 50 per Hue Bridge). Groups will also support a collection of Hue Bridges (instead of individual Hue light bulbs) to allow for control of multiple Hue Bridges with a single Event. These settings can be created or changed at any time when the app is paired to the Hue Bridge. If there is no connection, the application will not let the user view or change Traits of those controlled by the un-connected Hue Bridge. If the Hue Bridge itself cannot establish a connection to a particular Hue light bulb (or set of light bulbs) the application will not allow the user change the Traits of those light bulb(s). However, it will notify the user that those particular Hue light bulbs are not available due to the

Hue Bridge's failure to communicate with them. This could be due to, for example, the physical light switch that controls the socket the Hue light bulb is installed in is off or because the Hue light bulb is broken or malfunctioning.

While connected to a Hue Bridge the user will be able to control colors and behavior of the light bulbs. Changing colors can be done in one of two different ways. The first way is by picking a certain color from a color pallet and assigning that color to a specific bulb. The second is by picking colors from a picture that the user chooses from the device gallery or from a picture that has been taken in real time by the camera. While choosing a color the user will be able to specify the density and brightness of that color. Colors that were picked can be saved with a specified name as themes, and afterwards can be quickly accessed from the app to activate, modify, or delete.

Widget

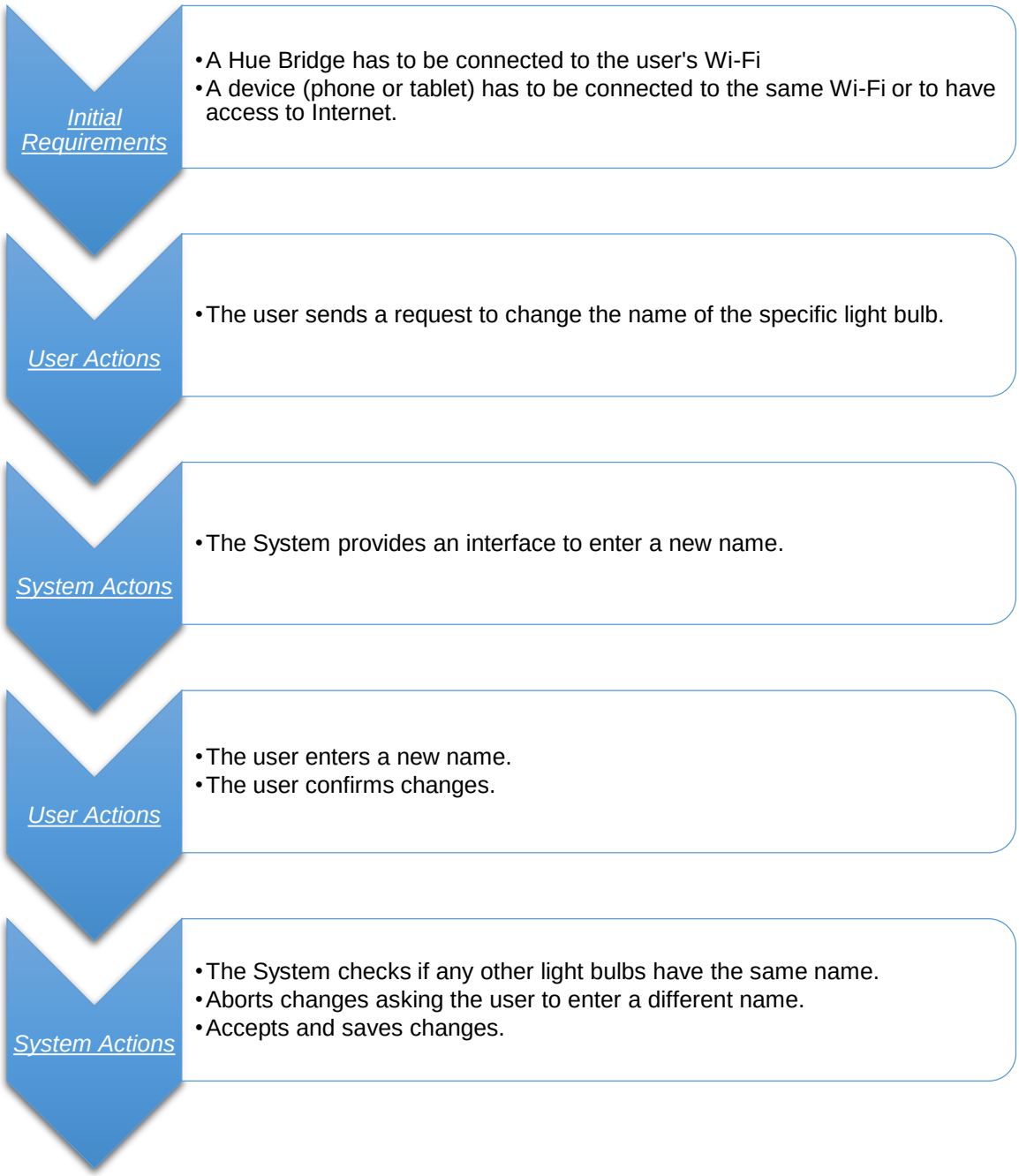
Bright Light Systems will also supply the user the option to add an interactive widget to their home screen. This widget will allow for the user to control the only currently connected bridge, which will be automatically detected / refreshed, with basic functionality such as on/off, bright/dim mode, etc.

USE CASE:

The following use cases describe the essential functionality of the app:

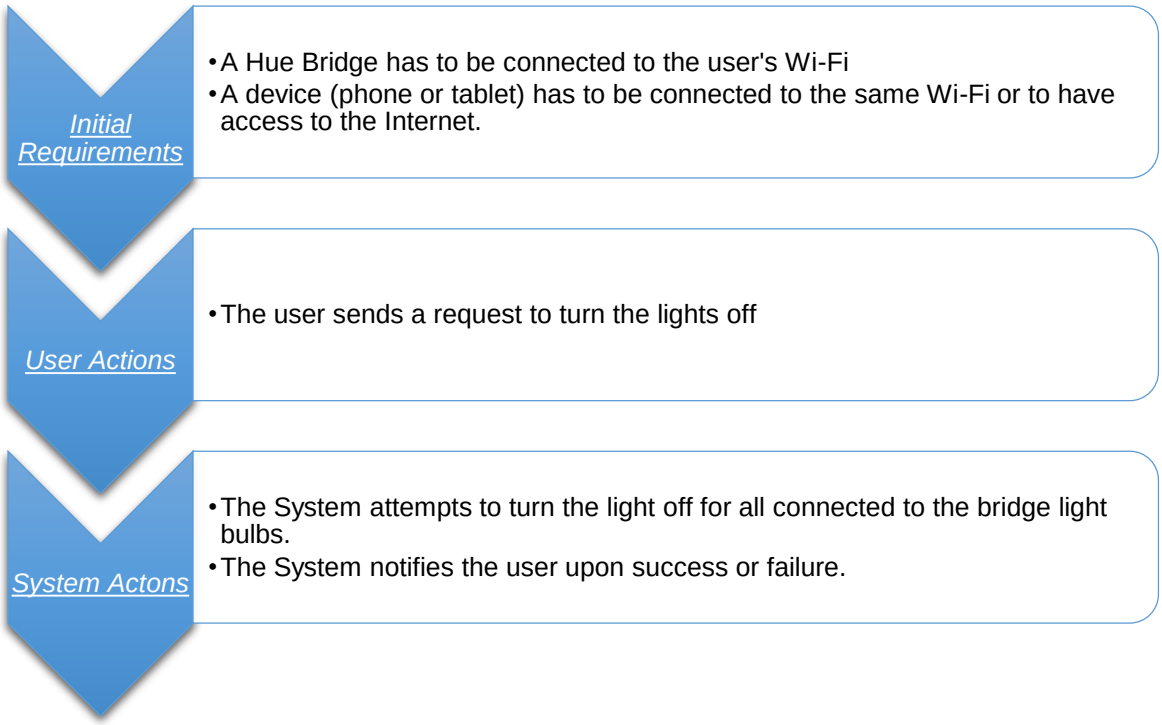
Use Case 1: Naming Light Bulbs	
Goal:	The user names/renames connected to the bridge light bulbs

Process:



Use Case 2: Turning off the Lights	
Goal:	The user turns the lights off

Process:



Use Case 3: Turning on the Lights	
Goal:	The user turns the lights on

Process:

Initial Requirements

- A Hue Bridge has to be connected to the user's Wi-Fi
- A device (phone or tablet) has to be connected to the same Wi-Fi or to have access to the Internet.

User Actions

- The user sends a request to turn the lights on using the previous session's settings, or by selecting a saved earlier trait.

System Actons

- The System attempts to turn the light on for the light bulbs connected to the bridge, and sets them to one of the following settings: default, previous session, or new session depending on the request.
- The System marks and presents to the user any light bulbs that are not connected as 'unreachable'.
- The System notifies the user upon success or failure.

Use Case 4: Creating a Trait

Goal:	The user creates a trait for light bulb(s)

Process:

Initial Requirements

- A Hue Bridge has to be connected to the user's Wi-Fi
- A device (phone or tablet) has to be connected to the same Wi-Fi only.

User Actions

- The user sends a request to create a new trait.

System Actons

- The System provides a list of available light bulbs, which will be used in the trait, and an interface to choose a color, saturtion, and a brightness of the light bulb(s).
- The System marks and presents to the user any light bulbs that are not connected to the Wi-Fi as 'unreachable'.

User Actions

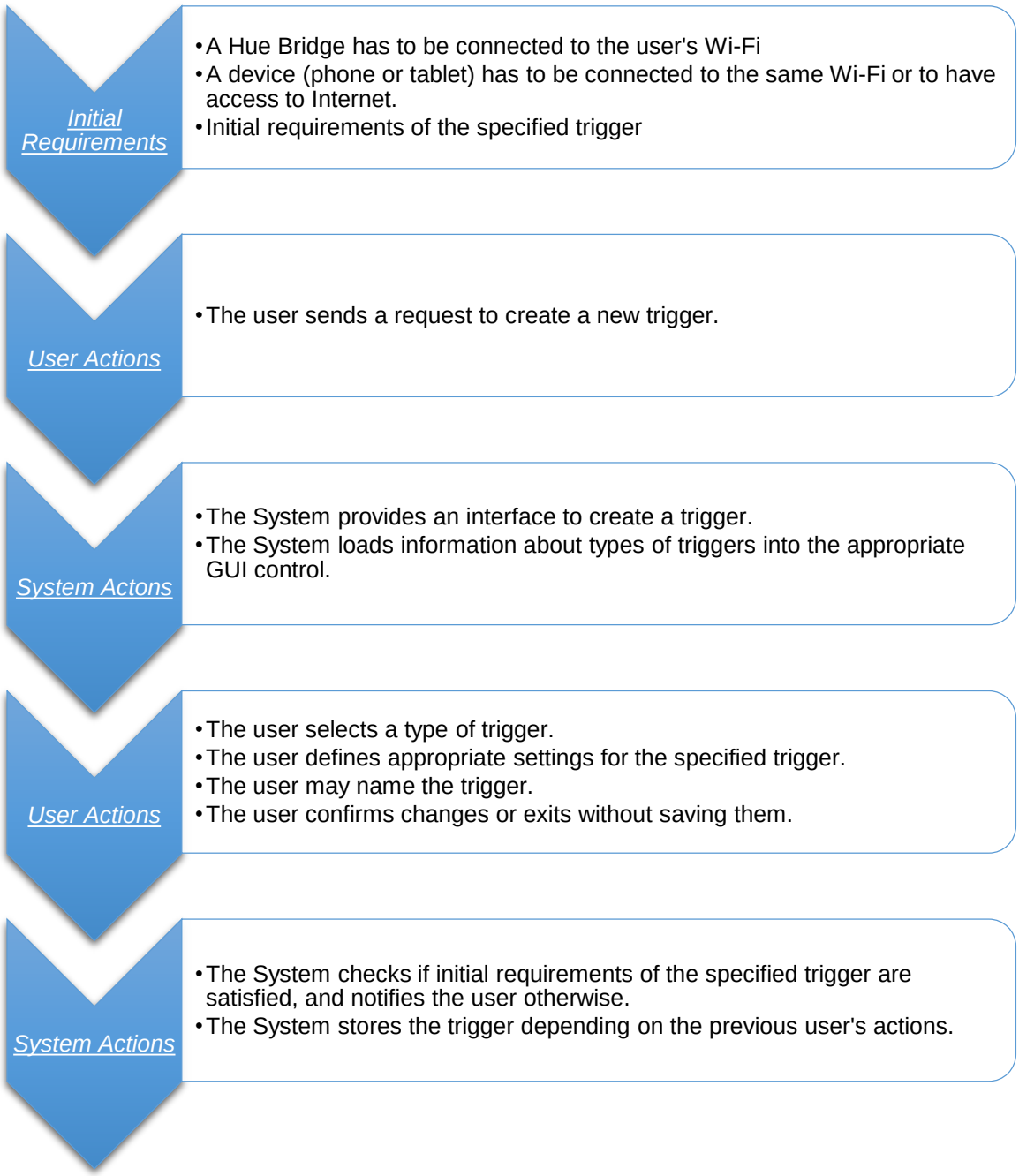
- The user may remove/add light bulb(s) from the list.
- The user may choose color, saturation, and brigtness for a single or multiple light bulbs.
- The user may choose to name the trait.
- The user saves and exits, or exits without saving changes.

System Actions

- The System presents changes to the user through the GUI controls and real time changes of light bulb(s).
- The System sets any changes as a current light setting.
- The System may store the trait for future use depending on previous user actions.

Use Case 5: Creating a Trigger	
Goal:	The user creates a trigger for an event.

Process:



Use Case 6: Creating an Event:

Goal:	The user creates an Event

Process:

Initial Requirements

- A Hue Bridge has to be connected to the user's Wi-Fi
- A device (phone or tablet) has to be connected to the same Wi-Fi only
- Information about stored earlier traits and triggers.
- Initial requirements of the specified trigger

User Actions

- The user sends a request to create an Event

System Actons

- The System provides an interface to define desired settings.
- The System loads information about created traits and triggers into apropriate GUI controls.

User Actions

- The user may select an already created trait from the list.
- The user may select an already created trigger from the list.
- The user may edit a trait or trigger on the fly and use it for the event.
- The user may create a new trait and trigger on the fly and use it for the event.
- The user may name the event and stored it for future use.
- The user may specify to start the event right after confirming changes
- The user confirms changes or exits without saving.

System Actions

- The System does not let the user confirm changes until both a trait and a trigger are specified.
- The System checks if the Initial requirements for the selected trigger are satisfied and notifies the user otherwise.
- The System may store the created event depending on the user's action.
- The System may immediately start the event depending on the user's action.

NON-FUNCTIONAL REQUIREMENTS

- Battery life and Performance

  Being an application for a mobile platform, this app should perform as efficiently as possible. On an “average” device this app shouldn’t exceed 15% of the battery usage. This is an important metric to look at because the feature set involved will require background services to listen for events. Additionally as an Android standard, any resources not currently being used by the application should be released upon completion of its task (i.e. camera, microphone, etc.) as to not conflict with any other applications / Android OS operation.

- Security

  The Philips Hue system has built in measures for security that this app will have to comply with these to function properly. Upon pairing to a Hue Bridge, the system will require the user to prove physical access to the system by pressing a button on the Hue Bridge; much like WPS technology implemented on many of today’s wireless routers. The main security goal for this project is to not allow unintended access to a Bridge or set of lights.

- Response Times

  The user experience must *feel* fast; this is often hard to measure in the case of an Android application. Events should be visibly “completed” within 1 second. In regards to the feel of the response time, Hue bulbs include transition times between events; this should be reduced from the stock value or user adjustable.

- Processing Times

  In many cases because of its core functionality, this app will require little processing time. One case where this is important, however, is ongoing services such as notification listening and music mode (and other triggers). Music mode should be able to buffer a section of the song to determine the BPM to set the light animation to. This “sampling” process should (attempt to) take no longer than 5 seconds before the lights are on beat and animating.

- Recovery

  As an app that will allow for extensive customization of their home, it is important that this information is able to be recovered. The app (in settings or where appropriate) should allow for the user to backup their user data pertaining to this app. This feature should build out and allow the user to export a file containing enough information to replicate the user’s current app state.