*Slack Technologies*

Cloud-based Team Collaboration Tool

Version 1.0

*Alexander R Ramsey*

Disclaimer: This document is generated for educational purposes only   
and does not officially represent Slack Technologies.

Revision History

|  |  |  |  |
| --- | --- | --- | --- |
| **Date** | **Version** | **Description** | **Author** |
| 04/03/2017 | 1.0 | Document inception | Alexander Ramsey |

Table of Contents

1. Introduction 4

1.1 Purpose 4

1.2 Scope 4

1.3 Definitions, Acronyms and Abbreviations 4

1.4 Overview 5

2. Overall Description 6

2.1 Use-Case Model Survey 6

3. Specific Requirements 8

3.1 Use-Case Reports 8

3.1.1 Access Via Web Browser 9

3.1.2 Create Account 9

3.1.3 Create Team Account 10

3.1.4 Join Existing Team Account 16

3.1.5 Revisit Team Account Cross Platform 17

3.1.6 Revisit Team Account Cross Platform – Mobile form factor 21

3.1.7 Post to a Channel/Topic 25

3.1.8 Create a Channel/Topic 26

3.1.9 Pin an Item to a Channel/Topic 27

3.1.10 Mention a Team Member 29

3.1.11 Mention a Channel/Topic 30

3.2 Supplementary Use Case Information 32

3.3 Use Case Diagram 32

4. Supporting Information 33

# Introduction

Slack Technologies envisions a better future for team collaboration with a cross platform, developer-open API, cloud based messaging centered tool. This tool allows teams to share information across messaging channels, host meetings, and integrate with other SaaS technologies. Their enterprise offering will allow the company to grow as an emerging leader in the cloud collaboration solution marketplace.

## Purpose

The purpose of the Slack Cloud-based Team Collaboration Tool is to allow teams of geographically separated teams of people to come together in one forum for seamless communication and collaboration.

Below are described nonfunctional and functional requirements. Nonfunctional requirements are inherent to the application and their development will not be defined as such in this requirements document. Functional requirements are defined by the use case model survey and use case reports in this document. Some nonfunctional requirements are met by specific functional requirements. For example, cross platform usage of the application is defined with the use of a mobile device.

Nonfunctional requirements include:

1. Cloud based
2. Cross platform
3. Asynchronous user experience
4. SaaS Integration

Functional use case requirements include:

1. Access via Web Browser
2. Create Account
3. Create Team Account
4. Join Existing Team Account
5. Revisit Team Account Cross Platform
6. Revisit Team Account on a Mobile form factor
7. Post to a Channel
8. Create a Channel/Topic
9. Pin an Item to a Channel/Topic
10. Mention a Team User
11. Mention a Channel/Topic

## Scope

Slack is a messaging suite for teams with data accessible via a back-end cloud web service and cross platform front-end. It will allow for teams to communicate via Team Accounts. Team Accounts are accessible via an easy to remember and custom set URL. A Team Account will provide teams with the ability to create Topic chat areas. Each Topic chat area will have pinned items. Each Topic chat area will allow users to mention others. Topics will be visible upon defined user roles. Topic notifications can be muted. Topics can integrate with other SaaS solutions to start a web meeting, share a gif image, or to share a document.

## Definitions, Acronyms and Abbreviations

Modern industry terms and abstract terms related to web chat programs are used throughout this SRS document. Brief definitions for these terms are provided as follows:

* Slack Technologies – represents an actual organization founded in 2013 with more than 700 employees. This company is head by Stewart Butterfield, a founding member of Flickr and former employee of Yahoo, Inc.
* Cross Platform – software technologies which operate seamlessly across multiple device form factors and operating systems.
* Cloud based, Cloud – term used to describe how a web application is hosted and accessible to the public across the broader internet.
* SaaS – Software as a Service, a cloud service model in which hosting, development, and support are provided for the end user.
* SaaS Integration – the interoperability of information or data between the chat program Slack, with other SaaS products.
* HTTP – Hypertext Transfer Protocol, a communications protocol used in the current World Wide Web ecosystem
* JavaScript – a currently popular, loosely typed, web language and technology ecosystem
* Front end – the client side scripting used to develop a user interface interpreted by web browsers installed to an operating system, often communicates to the back end via HTTP protocol requests
* Back end – the server side components which listen for front end requests and supply an HTTP response
* API – Application Platform Interface, a technical definition by which data is exchanged successfully by following a defined schema and access protocol among front end and back end web technologies.
* Asynchronous – term used to describe current web technologies wherein JavaScript calls are used to point to an API without browser page refreshes, thus improving the user experience.
* Team Account – accessible protected area that contains team members
* Topic – Also known as Channel, a chat channel pertaining to a specific subject. These terms are loosely mixed because of the topic driven nature of the channel.

## Overview

The following sections provide an overall description of the requirements covered in the specific requirements use case reporting area.

# Overall Description

Slack team collaboration tool extends upon former online chat systems with the capability of providing for multiple user’s access to shared, asynchronous, cloud based, multi-platform channels (aka Topics) to relay information. Slack extends this experience of chat by integrating other popular web services.

Topics can be customized to suit the needs of the team and its specific member breakdown. Important posts can be pinned to a topic for later reference. Users can be mentioned to grab attention to specific discussions in the channel. Documents can be shared amongst the team to improve organizational capability. Meetings can be generated to allow for greater face to face and voice collaboration.

## Use-Case Model Survey

|  |  |
| --- | --- |
| **Nonfunctional Requirements** | |
| **Requirement** | **Description** |
| Cloud based | Cloud based, Software as a Service (SaaS) employed application for easy setup and integration with existing organizational systems. |
| Cross platform | Information in the system should be accessible via a number of popular use devices on desktop and mobile form factors. |
| Asynchronous user experience | Asynchronous communication with web servers to improve communication speed and user experience. |
| SaaS Integrations | Easy integration with external, third party vendor SaaS software services to make work more fun and easy for teams. |

|  |  |
| --- | --- |
| **Use Case Actors** | |
| **Name** | **Role** |
| User | Any user of the system |
| Team Organizer | A user which initiates a Team Account. Inherits Team Member capabilities. |
| Team Member | A user which joins an existing Team Account |

|  |  |  |
| --- | --- | --- |
| **Functional Requirements** | | |
| **#** | **Use Case** | **Description** |
| 1 | Access via Web Browser | Any User should be able to access the system via a Web Browser |
| 2 | Create Account | Any User should be able to create an account. |
| 3 | Create Team Account | A Team Organizer should be able to create a team based account with a universal resource locator (URL). |
| 4 | Join Existing Team Account | A Team Member(s) should be able to join the existing team based account by invitation from a Team Organizer. |
| 5 | Revisit Team Account Cross Platform | A User should be able to revisit the team based account via the use of the generated URL. |
| 6 | Revisit Team Account Cross Platform – Mobile form factor | A User should be able to revisit the team based account via the use of the generated URL using a compatible mobile software package. |
| 7 | Post to a Channel | A Team Member should be able to post to a General topic channel. |
| 8 | Create a Channel/Topic | A Team Organizer should be able to create topic-based chat channels. |
| 9 | Pin an Item to a Channel/Topic | Any Team Member should be able to pin important information to a topic channel |
| 10 | Mention a Team Member | Any Team Member should be able to mention another Team Member |
| 11 | Mention a Channel/Topic | Any Team Member should be able to mention the channel topic to alert all team channel members |

# Specific Requirements

Use-Case Reports section provides detailed system operation pertaining to the 16 defined functional requirements use cases defined in the Use Case model survey.

## Use-Case Reports

### Access Via Web Browser

Summary: Any User should be able to access the system via a Web Browser

Preconditions: User should have a computer with internet access and a modern web browser installed.

Triggers: User accesses system.

Scenario:

|  |  |  |
| --- | --- | --- |
| Actor | System | Screen |
| (1) User visits <https://slack.com> in web browser |  |  |
|  | System returns sign up homepage. |  |

Postconditions: None.

### Create Account

Summary: Any User should be able to create an account.

Preconditions:

Triggers:

Scenario:

|  |  |  |
| --- | --- | --- |
| Actor | System | Screen |
| User (1) enters valid email address and (2) selects Get Started |  |  |
|  | System asks user for further input, (1) Join an existing team or (2) Create a new team. |  |

Postconditions:

### Create Team Account

Summary: A Team Organizer should be able to create a team based account with a universal resource locator (URL).

Preconditions: User should have a valid and accessible email address.

Triggers:

Scenario:

|  |  |  |
| --- | --- | --- |
| Actor | System | Screen |
| User (1) selects Create a new team option |  |  |
|  | System asks for confirmation code. System emails confirmation code to User supplied email address. |  |
| User retrieves code from email inbox. |  |  |
| User supplies valid code from email inbox message. | System confirms email address is valid. System asks for additional User information including:   * First name * Last name * Username preference (autofilled) * Email notifications preferences |  |
| User supplies information. |  |  |
| User selects to Continue to Password |  |  |
|  | System displays password setup screen. |  |
| User supplies valid password and selects Continue. |  |  |
|  | System prompts for Team information with the following options:   * Team usage * Size of team |  |
| User selects options for Team:   * School * 1-10 users   User selects Continue to Group Name. |  |  |
|  | System prompts for Group Name. |  |
| User supplies desired Group Name. |  |  |
|  | System responds with team name availability. |  |
| User continues to Create Team |  |  |
|  | System prompts user to Agree to Customer Terms of Service and Privacy Policy |  |
| User Selects I Agree, after reviewing terms. |  |  |
|  | System prompts for email confirmation code. |  |
| User enters correct confirmation code. |  |  |
|  | System provides form for inviting team member users. |  |

Postconditions: System persists created Team Account with specified URL. User actor becomes Team Organizer actor.

Alternative Path:

User supplies invalid inputs during sign up process.

|  |  |  |
| --- | --- | --- |
| Actor | System | Screen |
|  | System prompts for supplied email confirmation code or other required field during sign up process. |  |
| User supplies invalid requested information. |  |  |
|  | System provides asynchronous validation error and does not proceed to next step. |  |

### Join Existing Team Account

Summary: A Team Member(s) should be able to join the existing team based account by invitation from a Team Organizer.

Preconditions: Slack Create Team process performed.

Triggers: Team Organizer has generated an account and a Slack Team.

Scenario:

|  |  |  |
| --- | --- | --- |
| Actor | System | Screen |
| Team organizer enters email address of Team Member. |  |  |
|  | System delivers invitation email message to Team Member. |  |

Postconditions: Team Member actor receives email message.

### Revisit Team Account Cross Platform

Summary: A User should be able to revisit the team based account via the use of the generated URL.

Preconditions: Team Member provided with invitation from Team Organizer. System successfully sends email invitation message.

Triggers: Team Organizer sends invitation. Team Member joins team.

Scenario:

|  |  |  |
| --- | --- | --- |
| Actor | System | Screen |
| Team Member receives email invitation message. |  |  |
| Team Member clicks Join Team. |  |  |
|  | System responds by displaying User Sign Up process.  The following form fields are required:   * First Name * Last Name * Username |  |
| Team Member enters requested details |  |  |
|  | System responds with Terms and Conditions message. |  |
| Team Member agrees to Terms and Conditions message. |  |  |
|  | System redirects user to Slack Team Account homepage. |  |
|  | System sends email confirmation to User Team Member including Team Account URL. |  |
| Team Member visits Team Account URL in alternative platform web browser. |  |  |
|  | System prompts for authenticated login. |  |
| Team Member enters correct account username and password. |  |  |
|  | System returns Slack Team Account home page. |  |

Postconditions: Team Account URL is shared with Team Members for access in other platforms and form factors.

Alternative Path:

A. Team Member supplies invalid Team Account URL.

|  |  |  |
| --- | --- | --- |
| Actor | System | Screen |
| Team Member visits incorrect Team Account URL in alternative platform web browser. |  |  |
|  | System rejects request. |  |

B. Team Member supplies invalid Authentication details.

|  |  |  |
| --- | --- | --- |
| Actor | System | Screen |
|  | System prompts for Authenticated login. |  |
| Team Member enters invalid Authentication details. |  |  |
|  | System rejects request. System prompts user for corrected Authentication details. |  |

### Revisit Team Account Cross Platform – Mobile form factor

Summary: A User should be able to revisit the team based account via the use of the generated URL using a compatible mobile software package.

Preconditions: User has installed the Slack mobile application on their device. User is a Team Member or Team Organizer and knows their Team Account URL and Slack Account Authentication details.

Triggers: User opens Slack application on their mobile device.

Scenario:

|  |  |  |
| --- | --- | --- |
| Actor | System | Screen |
| Team Member accesses System. |  |  |
|  | System returns a welcome screen with the option to Sign In or Create a Team. | C:\Users\Alex\AppData\Local\Microsoft\Windows\INetCacheContent.Word\Screenshot_20170305-160534.png |
| Team Member selects the Sign In option. |  |  |
|  | System prompts user for Team Account URL. |  |
| Team Member enters Team Account URL. |  | C:\Users\Alex\AppData\Local\Microsoft\Windows\INetCacheContent.Word\Screenshot_20170305-160547.png |
|  | System prompts for Email Address authentication detail. |  |
| Team Member enters Email Address authentication detail. |  | C:\Users\Alex\AppData\Local\Microsoft\Windows\INetCacheContent.Word\Screenshot_20170305-160557.png |
|  | System confirms email address. System prompts Team Member for Password authentication detail. |  |
| Team Member supplies valid Password authentication detail. |  | C:\Users\Alex\AppData\Local\Microsoft\Windows\INetCacheContent.Word\Screenshot_20170305-160609.png |
|  | System displays General chat channel to the Team Member. | C:\Users\Alex\AppData\Local\Microsoft\Windows\INetCacheContent.Word\Screenshot_20170305-160614.png |

Postconditions: Team Member mobile access actions persist to the system across all form factors in real time.

### Post to a Channel/Topic

Summary: A Team Member should be able to post to a General topic channel.

Preconditions: Team Member is logged into the system and has navigated to the General channel.

Triggers: Team Member accesses the system. Team member enters a comment in the General channel.

Scenario:

|  |  |  |
| --- | --- | --- |
| Actor | System | Screen |
| Team Member accesses the system via any form factor. |  |  |
|  | System displays General chat channel user interface. |  |
| Team Member enters a comment into the text field for the General channel and submits by pressing the Enter key. |  |  |

Postconditions: Team Member comment saves to the system and is displayed to all other Team Members and Team Organizers of the Team Account without browser refresh.

### Create a Channel/Topic

Summary: A Team Organizer should be able to create topic-based chat channels.

Preconditions: Team Organizer has access to the system.

Triggers: Team Organizer presses create channel button.

Scenario:

|  |  |  |
| --- | --- | --- |
| Actor | System | Screen |
| Team Organizer selects create channel button. |  |  |
|  | System displays Create Channel pane with the following fields:   * Public or Private * Name * Purpose * Invitations |  |
| Team Organizer enters required fields and selects Create Channel. |  |  |
|  | System generates a new channel and provides options for Team Members to join the channel. |  |
| Team Organizer selects Got It to proceed to the Channel. |  |  |
|  | System displays channel posts with the set Purpose. |  |

Postconditions: Channel is saved, shared, and available to Team Members.

### Pin an Item to a Channel/Topic

Summary: Any Team Member should be able to pin important information to a topic channel

Preconditions: Team Member has access to the channel.

Triggers: Team Member triggers the pin post menu button.

Scenario:

|  |  |  |
| --- | --- | --- |
| Actor | System | Screen |
| Team Member selects post options menu. |  |  |
|  | System displays post options menu. |  |
| Team Member selects Pin to #general menu option. |  |  |
|  | System displays Pin confirmation message. |  |
| Team Member selects, Yes, Pin this Message |  |  |
|  | System Pins the selected post. |  |
|  | System saves Pinned item in the Pinned Items sidebar menu. |  |

Postconditions: Message is pinned to the channel and is accessible via a System Pinned Items sidebar menu area.

Alternative paths:

Team Member cancels the Pin action during confirmation message.

|  |  |  |
| --- | --- | --- |
| Actor | System | Screen |
|  | System displays Pin confirmation message. |  |
| Team Member selects Cancel. |  |  |
|  | System hides dialog and menu. System returns to channel. |  |

### Mention a Team Member

Summary: Any Team Member should be able to mention another Team Member

Preconditions: Team Member is a member of a channel with multiple Team Members.

Triggers: Team Member uses predefined syntax for mentioning other team members in the chat field.

Scenario:

|  |  |  |
| --- | --- | --- |
| Actor | System | Screen |
| Team Member enters the following formatted text into the chat field:  @ |  |  |
|  | System responds asynchronously with a list of recommended values following the special character. |  |
| Team Member selects desired Team Member to Mention. |  |  |
|  | System fills the Chat field with the correct syntax to mention other Team Member |  |
| Team Member enters comment following the Mention. |  |  |
| Team Member presses Enter key on their keyboard. |  |  |
|  | System posts comment to the channel. System issues Mentioned Team Member a notification of the comment. |  |

Postconditions: Posted comment saves in the channel. Mentioned Team Member receives an application notification message of the mention.

### Mention a Channel/Topic

Summary: Any Team Member should be able to mention the channel topic to alert all team channel members

Preconditions:

Triggers:

Scenario:

|  |  |  |
| --- | --- | --- |
| Actor | System | Screen |
| Team Member enters the following formatted text into the chat field:  @ |  |  |
|  | System responds asynchronously with a list of recommended values following the special character. |  |
| Team Member selects desired option to mention the channel. |  |  |
|  | System fills the Chat field with the correct syntax to mention the Channel. |  |
| Team Member enters a comment following the mention. |  |  |
| Team Member presses Enter on their keyboard. |  |  |
|  | System posts comment to the channel. System issues a notification to all channel Team Members. |  |

Postconditions: Comment is saved in the channel. All channel Team Members are notified.

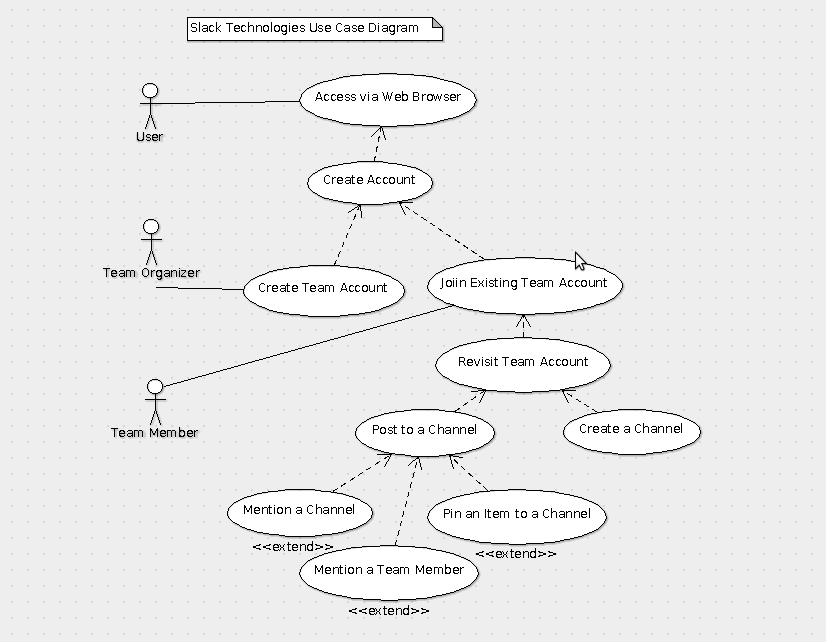
## Supplementary Use Case Information

Throughout the use of the application, Alternative Paths involving form field validation may occur. These alternative paths occur in an asynchronous manner, meaning that they respond instantly to user input. If user input is not sufficient for validation to pass, then the system will not allow the user to continue to the next step. A validation error message is displayed. To reduce the size of this document, many of these noncritical instances of validation are supplemented with the following generalized Alternative Path:

User supplies invalid inputs throughout application usage.

|  |  |
| --- | --- |
| Actor | System |
|  | System prompts for required field and/or response. |
| User supplies invalid requested information needed to proceed to following step. |  |
|  | System provides asynchronous validation error and does not proceed to next step. System remains at current browser page without refresh until session timeout occurs. |

## Use Case Diagram



# Supporting Information

Rational Unified Process Documentation. 1999-2007. Accessed via http://sce.uhcl.edu/helm/rationalunifiedprocess/index.htm

Slack Technologies. 2017. Accessed via http://www.slack.com