

Improving User Experiences for Wiki Systems

A Project Report

Presented To

Dr. Chris Pollett

Department of Computer Science

San José State University

In Partial Fulfillment

Of the Requirements for the

Class CS297

By

Parth Amrutbhai Patel

May 2022

ABSTRACT

Wiki systems are kind of web application that allows users to collaboratively manage its content. Such systems enable users to read and write information in form of web pages and share media items like videos, audios, books etc. Users can communicate and discuss among themselves by forming groups and discussion boards. Yioop is an open-source web portal with search engine, a wiki system and discussion board. The final aim of this project is to establish a mechanism for recommending various media items to users of wiki systems like Yioop which will greatly enhance their experience of using a wiki system. The goal for this semester is to gain knowledge about Yioop's functionalities and code base by introducing new features which will improve user experiences. The task for this semester was divided into four key deliverables. The first deliverable is implementation of emojis for the direct messaging system in Yioop. The second deliverable is creation of UI testing framework using Selenium technologies. The third deliverable is implementation of credits conversion into US dollars using Stripe APIs. The fourth deliverable is refactoring of code implementing Hash2Vec approach for recommending groups and threads according to Yioop's coding guidelines ensuring its successful integration into Yioop's codebase.

Keywords – Wiki Systems, Yioop, Selenium, Stripe, APIs, Web Pages, Direct-Messaging, Credits, Hash2Vec, Recommendation, Groups, Thread

TABLE OF CONTENTS

I. INTRODUCTION	1
II. DELIVERABLE 1: EMOJI PICKER TOOL FOR DIRECT MESSAGING ...	3
III. DELIVERABLE 2: CREATE USER INTERFACE TESTING PROJECT	5
IV. DELIVERABLE 3: CREDITS CONVERSION TO US DOLLARS	7
V. DELIVERABLE 4: REFACTOR ANIRUDH'S HASH2VEC CODE	9
VI. CONCLUSION	11
REFERENCES.....	12

I. INTRODUCTION

Wiki system provides very quick and efficient way of creating, consuming and managing of information through web pages. It allows collaboration among users by providing facilities of creating groups and discussion boards for common communication among the group members. Yioop an open-source web portal is one such wiki system which additionally has a search engine functionality. The users can create groups and configure rules of joining the group and the access level of the group members. Every group has related wiki pages which the users can view and edit. Moreover, every group has a discussion board where users can start new threads and post comments to the existing threads. The access level restricts group members activities within the group for example whether members can edit wiki pages, can start new discussion threads, etc.

For any wiki system it is important to establish positive user experiences. The main elements of any wiki system that affect user experience are the User Interface (UI) and the features or functionalities provided to the user. Yioop provides few configurations through which users can customize certain part of the UI as per their taste. However, there are certain areas in Yioop where new features can be added for enhancing user experiences.

So, the goal for this semester project is to extend Yioop for enhancing user experiences primarily by implementing couple of new features in Yioop and secondarily by developing a UI testing framework to detect UI defects before the users report it thus ensuring maximum positive user experience for Yioop.

The rest of the report is organized as follows: Section I describes how an Emoji Picker tool was added to the Yioop's direct messaging activity. Section II discusses about a need for new UI testing mechanism for Yioop and how it is achieved. Section III presents a functionality of converting unused credits back to US dollars. Section IV briefs about the former master student Anirudh Mallya's work of implementing Hash2Vec approach in Yioop's recommendation job and how this project refactored his code. Finally, the conclusion section summarizes outcomes for this semester and future work for the next semester.

Pre-requisites: **Yioop** is used for all the deliverables of this project. It can be downloaded from: <https://www.seekquarry.com/p/Downloads>.

II. DELIVERABLE 1: EMOJI PICKER TOOL FOR DIRECT MESSAGING

Direct Messaging is a quick communication between two parties allowing them to exchange text messages and text files up to certain size. Yioop also provides direct messaging feature where a user has ability to communicate privately with other users. However, the current implementation does not support feature of using emojis along with text in the messages exchanged. Given the popularity of emojis, it is absolute necessity for any direct messaging service to support it. The motive for this deliverable is to give users a feel of modern messaging application while using Yioop's direct messaging service and allow to express their emotions more clearly by using emojis in messages.

So, the goal of this deliverable is to implement an Emoji Picker tool into Yioop's direct messaging activity. In order to implement this feature, it was necessary to gain familiarity with the Unicode emoji characters and their code points. Every emoji is a Unicode character which has either a specific code point or a sequence of code points. To display an emoji on a web page one could use combination of Hyper Text Markup Language (HTML) codes equivalent to Unicode code point. For example, $U+1F600$ is code point for one of the smiley emoji, using its HTML equivalent code $\&\#xF600$ in a web page will allow web browser to display the emoji character instead of plain text.

Based on this information the Emoji Picker tool was designed without any dependency on external code libraries. Additionally, the code implementation is modular which allows to quickly integrate this tool to different text exchanging

functionalities in Yioop. The tool is designed to almost match the look and functionality of emoji picker tools in any modern direct messaging applications like WhatsApp.

Currently there are more than 1500 emojis supported by the tool, divided into 8 different categories allowing users to easily locate an emoji. Moreover, implementation also supports feature of replacing textual shortcuts of emojis in the messages with the corresponding emoji character. Fig. 1 shows the emoji picker tool, when a user hover on any emoji a label is shown which displays the name of the emoji along with the textual shortcut to be used in messages to replace it with emoji after sending the message.

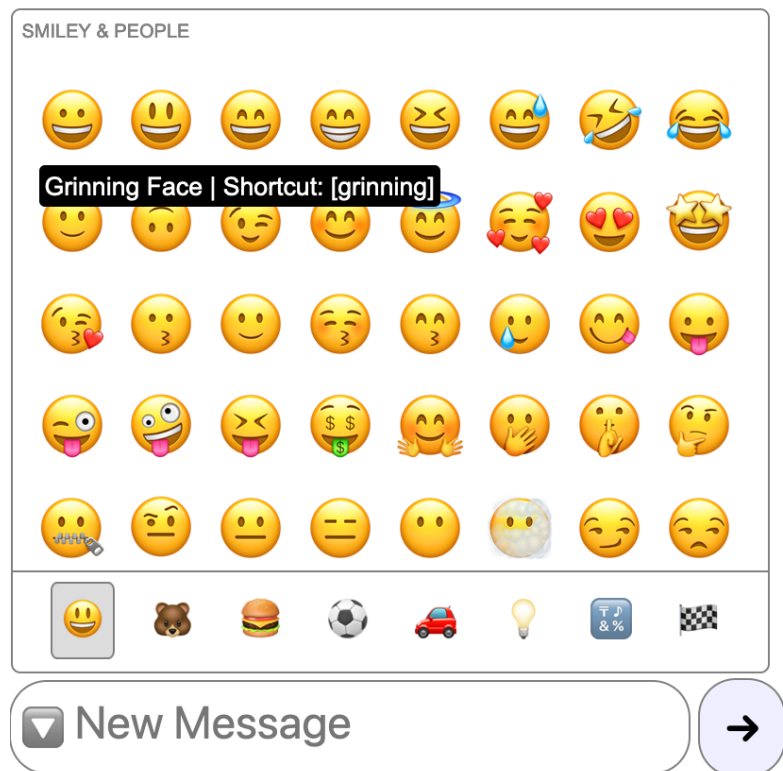


Fig 1. Emoji picker tool

III. DELIVERABLE 2: CREATE USER INTERFACE TESTING PROJECT

UI is the primary way through which a user can interact with any web application. To ensure rich user experience for a wiki system, a suite of UI test cases is always recommended. The purpose of these test cases is to simulate interaction of actual users with the system and ensure that every UI element works in the way it was designed. The failure of any test case denotes some defect with the UI elements being tested by it and requires immediate action to fix the issue.

The existing UI testing mechanism in Yioop is implemented using PhantomJS, which is an open-source JavaScript project. The main issues with current testing are outdated and limited test cases and PhantomJS project was discontinued by the owner. The motive for this deliverable is to address these issues and establish a new testing mechanism that can ensure maximum coverage of Yioop's UI so that any issues can be detected and fixed before users encounter them.

So, the goal for this deliverable is to create a UI testing tool for Yioop. The outcome of this deliverable is a separate project based on Selenium with node.js programming language which can test the Yioop's UI in Firefox and Google Chrome web browsers for desktop and mobile version of Yioop. Selenium is an open-source UI testing framework which provides various drivers for different web browsers and language bindings for different programming languages to use it.

The project currently has test cases for different UI activities like sign in, create, join and leave groups, start new threads in a group’s discussion board, etc. The project currently runs in headless mode meaning it will run test cases in the browsers without rendering the UI. When the test cases are executing, screenshots are captured at certain point which can help to take a glance at the UI in case of failure of any test cases.

A shell script was developed which ensures all the requirements of testing project are met before executing the test cases. The shell script checks whether Firefox and Google Chrome web browsers are installed on the system, downloads the latest copy of Yioop on the system and installs the required node packages for the project. Fig. 2 shows part of HTML report which is generated after all the test cases are executed. It shows the result of every test cases along with the screen shots captured. In order to completely automate the testing process, a cron job can be created which can execute the script after certain time intervals.

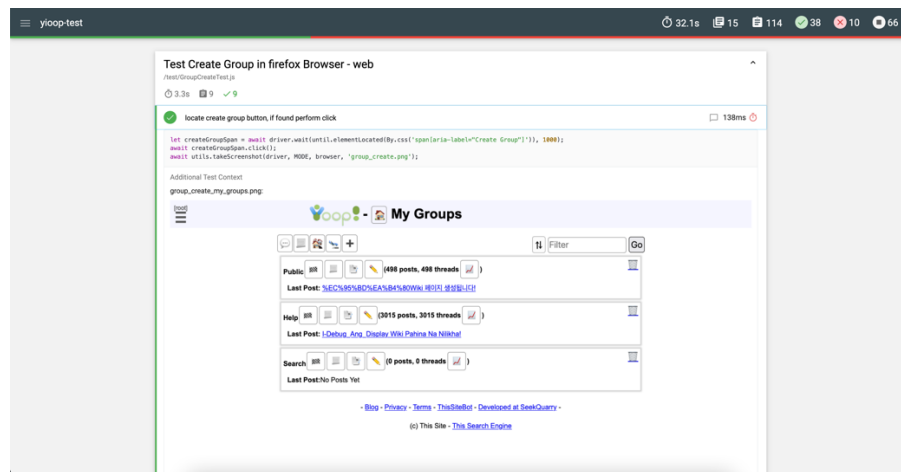


Fig 2. HTML report showing test cases result

IV. DELIVERABLE 3: CREDITS CONVERSION TO US DOLLARS

Yioop has a keyword advertisement facility which allows users with business roles to create text-based advertisement campaign for certain number of days. Whenever a user uses Yioop search engine then the appropriate advertisement campaign will be displayed based on the words in search query. In order to start an advertisement campaign user needs to bid for the keywords using *credits*. A credit is worth one US penny and user can purchase it using credit cards in a bundle of 1K, 2K, 5K and 10K credits. Users can also utilize credits as a fee for joining certain groups.

The existing functionality does not allow users to convert their unused credits back to US dollars. The motive behind this deliverable is that users can purchase credits only in a fixed bundle sizes as mentioned above. They might end up with unused credits and want to convert them back into money. So, the goal of this deliverable is to add support for redeeming credits back into US dollars. This implementation leverages Stripe's Connected Account services which allows to create Stripe accounts for the users. Stripe is a company that provides services to businesses for processing online payments.

Fig. 3 shows the new UI for Manage Credits activity. Purchase Credits tab has the existing functionality of purchasing the credits using bank cards. Redeem Credits tab allows users to redeem the credits after they have setup the Stripe connected account. Users can redeem credits by providing details of their debit card which are handled by Stripe only.

Manage Details tab allows the user to setup their Stripe connected account. Using this tab users will be redirected to Stripe portal where they are required to provide the verification details for using Stripe services. Users do not have direct access to their account, they can only access it through Yioop whenever some required information for Stripe account is missing. If a user does not provide any required information, the credit redeem functionality for such user will be blocked until the issues with Stripe account are resolved.

Balance: 5000 credits

Credit Transactions + Row 0 to 2 of 2 Show 50

[Purchase Credits](#) | [Redeem Credits](#) | [Manage Details](#)

Redeem Credits ?

Quantity:

Debit Card Number:

CVC:

Expiration: /

Using the Redeem button credits the above card the Quantity field's amount in US dollars and agrees to the [Program Terms](#).

Type	Amount	Date	Total
Starting Balance	0	Thu, 28 Apr 2022 22:18:26 -0700	0
Buy Credits	5000	Thu, 28 Apr 2022 22:27:25 -0700	5000

Fig. 3 Yioop Manage Credit activity

When a user redeems credits successfully, equivalent US dollars amount calculated considering Stripe fees and Yioop policy, will be transferred to the user's Stripe connected account. Afterwards Stripe will handle the process of transferring that amount to the user's debit card which was provided while redeeming the credits.

V. DELIVERABLE 4: REFACTOR ANIRUDH'S HASH2VEC CODE

When the user base for a wiki system increases, the amount of wiki pages and discussion boards also increases greatly. This makes it difficult for the users to find information as per their needs and tastes. To handle such situation many web applications leverage use of Machine Learning backed recommendation systems. Recommendation systems process the history of user interaction in the system and generates tailored view of information presented to the user.

Yioop also has a recommendation job that generates recommendation of groups and discussion board threads using term frequency and inverse document frequency technique. This technique was inadequate and generated ineffective recommendations. To improve the accuracy of recommendations, a former student Anirudh Mallya proposed using Hash2Vec approach during his master project. Hash2Vec method generates vectors to represent words which embeds the similarity with other words in the document. Anirudh's implementation did not follow Yioop's coding guideline which caused issues while integrating it into Yioop's codebase.

The motive for this deliverable is to gain familiarity with the Yioop's existing recommendation system and the internal data model for it which will significantly help to make decisions towards the final project for the next semester. So, the goal of last deliverable for this semester is to refactor Anirudh's Hash2Vec code to make it complaint with Yioop's coding guidelines.

Yioop has specific way to create or update databases tables by creating database version upgrade functions and changing the database version in config file to match that function version. Anirudh's code was also having minor issues with generating data corpus for generating vectors and with calculating similar words for a given word.

The outcome of this deliverable was refactored code that was correctly following coding guidelines. Along with that code was documented as per Yioop's guidelines which makes it easy for other readers to apprehend it and it improved the efficiency of calculating similar words by leveraging built in min heap data structure. The refactored code can now be successfully integrated into Yioop codebase without any issues.

VI. CONCLUSION

For any wiki systems like Yioop it is important to ensure a total positive experience for its users. In this semester, this project extended functionalities of Yioop to improve the user experiences. An Emoji Picker tool was developed for direct messaging activity which allows users to use emojis while sending messages. To ensure proper UI, a separate project using Selenium and a shell script was created which automated the UI testing process. Credits conversion functionality was implemented in Manage Credits activity which allows users to redeem unused credits back to US dollars. At last, Yioop's recommendation job was explored and Anirudh's code for Hash2Vec approach was refactored to ensure its successful integration into Yioop codebase.

The last deliverable will serve as the starting point for next semester work. Various research papers on collaborative recommendation strategies were explored during this semester. The final aim of this project is to establish mechanism for recommending media items like videos, books, magazines, etc. to the users of Yioop. The project will also focus to introduce functionality of posting tags in groups discussion board which can be used to increase accuracy of recommending groups and threads. This will help to present tailored information to the users, customized as per the user's history and taste.

REFERENCES

- [1]. M. Chiang, *Networked Life*. New York, NY: Cambridge University Press, 2012.
- [2]. F. Xue, X. He, X. Wang, J. Xu, K. Liu and R. Hong, "Deep Item-based Collaborative Filtering for Top-N Recommendation," *ACM Transactions on Information Systems*, Aug 2018.
- [3]. J. B. Schafer, D. Frankowski, J. Herlocker and S. Sen, "Collaborative Filtering Recommender Systems," *The Adaptive Web, Lecture Notes in Computer Science*, vol 4321. Springer, Berlin, Heidelberg.
DOI:https://doi.org/10.1007/978-3-540-72079-9_9.
- [4]. X. Li, D. Li, "An Improved Collaborative Filtering Recommendation Algorithm and Recommendation Strategy," *Mobile Information Systems*, vol. 2019, Article ID 3560968, 11 pages, 2019.
DOI:<https://doi.org/10.1155/2019/3560968>.
- [5]. <https://unicode.org/emoji/charts/full-emoji-list.html>
- [6]. <https://stripe.com/docs/api>