Project Status Report Team1

<Book Exchange>

Submitted: < April 2, 2020>

|  |  |
| --- | --- |
| **Prepared By:** | Machida Hiroaki |
| **Date Prepared:** | April 2, 2020 |

**Accomplishments Since Last Report**

* **Server Monitoring (Hiroaki)**

Developers monitor the free memory and disk of the application server through the internet (<http://team1.work/mrtg/>). That is because the free-tier t2.micro EC2 instance[1] havs only 1GB memory and 8GB disk space, and Xvfb[2], a display server, for Cypress[3] automated integration test needs around 2GB memory and npm[4] packages needs around 1GB disk space. MRTG[5] was installed by apt-get and configuration files and crontab are set up. Also, Nginx[6] route was modified. Installation done.

* **Whole user interface improvement – menu and transaction flow (Hiroaki)**

Users move around pages on menu and know the transaction status from transaction flow visualization. Hiroaki chose a Material-UI template[7] for menu. Only menu implemented. Transaction flow to be done.

* **Whole user interface improvement – overall design (Amol)**

Overall layout and fields in form shall be improved for better user experience. Amol is working on design.

* **Category Detection (Hiroaki)**

The category of book is automatically detected. Hiroaki got an AWS educate program[8] account, set up an EC2 Linux instance, and installed BERT-SQuAD[9][10][11] for an natural language processing API. The Elastic Block Store[12] is increased from 8GB to 16BG and a 5GB swapfile created for installing Pytorch[13]. A Elastic IP[14] is created and assigned to the EC2 instance and the server was launched. Also, an asynchronous call of API is added to the application. The API call is not so stable because it takes around 450 sec due to performance problem. Implementation done.

* **Category search (Shenyuan + Peiyan)**

Users search books by status and category. Shenyuan and Peiyan implemented. Since no complex extraction condition is available for firebase realtime database, the whole schema of book is taken into the application and extraction is done on the application. Implementation done.

* **Paypal payment (Shaurya)**

The application should keep the status on whether a user already makes a payment to another user. Paypal.me[15] will be used. Shaury is working on implementation.

* **Barcode reader (Shenjie)**

ISBN, that is a unique identification for a book, should be automatically input by uploading the bar code of a book. Shenjie successfully feed the camera data stream into the screen with Dynamsoft Barcode Reader SDK[16]. Not done yet.

* **Event notification (Hiroaki)**

Notifications for new events should be shown on the dashboard so that users can know what steps they need to take next. Hiroaki implemented event notification by adding a “notifications” schema on the firebase and and adding information after events. Implementation completed.

* **Iteration 2 Presentation (Hiroaki)**

Iteration 2 presentation and status report. Hiroaki made several changes of format based on the feedback for iteration 1 and grading rubric as follows. User stories listed and each user story explained in detail. Functionalities for next iteration is put at the end of presentation. Also, wide set of resources is listed up.

**Plans to do for Next Report**

* **Remaining of Iteration 2 (All)**

Remaining of iteration 2 needs to be completed.

* **Integration Test (All)**

Integration tests[17] for each implementation should be created by Cypress. The initial integration test is already implemented, so we need to expand them.

* **Return Book (Hiroaki + Shaurya)**

Users can request refunds if there are issues on the book after receiving the book. The user requests refunds on the application and the application should keep track of the status. First, screen design is needed.

* **Auto Complete Book Cover (Shenyuan + Shenjie)**

Application automatically downloads images of book covers from the internet or cache on the application and shows it to users. First, we need to decide how to get images.

* **Recommendations Engine (Amol + Peiyan)**

Recommendations engine is a feature that allows users to receive recommendations of new arrival of books based on their interest area. The user dashboard would have a space of recommendations’ from the system that match the user’s interests. User’s interest would be captured through user - preferences. First, screen design is required.

**Key Issues, Risks & Concerns**

|  |  |  |  |
| --- | --- | --- | --- |
| **Item** | **Action/Resolution** | **Responsible** | **Completion Date** |
| [Risk] Dev env setup | Not everyone set up dev env  [Action] Set up dev env in face-to-face  [Result] Everyone has set up the dev env. | Ding Shenjie | 2/25 |
| [Risk] Member skills | Not everyone familiar with React & Redux.  **[Action] Follow up questions from members frequently on Slack.** | Sinha Shaurya |  |
| [Risk] High workload functions | It’s very likely that there will be conflicts among functions for high workload development.  **[Action] In advance, write user stories, screen drafts, and db schemas.** | Each member |  |
| [Issue] High workload outside class | Some members have high workload outside of class and cannot proceed tasks.  **[Action] Made project rule and follow them.**  -No-show: agree things without someone not attending unless notified.  -Over-due task: change task owner.  -Over-due feedback: not reflected in principle. | Machida Hiroaki |  |
| [Issue] Coronavirus pandemic | Because of the coronavirus pandemic[18], we are no longer able to do meetings in person. Difficult to do pair programming and good communication. Implementation progress is not good.  **[Action] Reschedule and change the scope after iteration 2.**  [Comment]  We saw this as a risk, but changed to an issue, since barcode reader and paypal payment are not completed by iteration 2. That is because we could not support each other in face-to-face. One action Shenyuan took to mitigate the issue is to share issues by video recording with mobile phone and ask help. | Machida  Hiroaki |  |

**Status**

|  |  |  |
| --- | --- | --- |
| **Overall** | **Y** | Not all iteration 2 targets were completed. That is partially because we were not able to have face-to-face meetings because of corona virus pandemic. However, the basic user story, that users sell & buy books, was already implemented on iteration 1, so the impact to the whole project is limited. |

|  |  |  |
| --- | --- | --- |
| Schedule | **R** | Only server monitoring, category detection, category search, part of user interface improvement and event notification completed. The other part of user interface improvement, Paypal payment and barcode reader were not completed and put into iteration 3. See “Accomplishments Since Last Report“ for details. |
| Resources | **Y** | Team members not utilized. Because of corona virus pandemic, some members could not have face-to-face support. Also, one member had a common cold on the spring break and flied back to home country because of semi lockdown. |
| Scope | **Y** | Need to change the scope on iteration 3. As mentioned on the schedule section, some implementations are put from iteration 2 into iteration 3. However, workload may be too high for one iteration. After iteration 2, we need to discuss internally how much resource available and change the scope on iteration 3. |

**Schedule**

|  |  |  |  |
| --- | --- | --- | --- |
| **Stage (Sprint)** | **Target Completion** | **Revised Completion** | **Comments** |
| **Iteration1 (Must-have)** | 3/5 | 3/5 | Done |
| Integration test automation framework setup | 2/20 | 2/20 | Hiroaki Done |
| Staging environment setup | 2/20 | 2/20 | Hiroaki Done |
| Continuous integration environment setup | 2/20 | 2/20 | Hiroaki Done |
| Minimum user story | 2/20 | 2/20 | Hiroaki & Shaurya Done |
| User review for user | 3/4 | 3/4 | Amol Done |
| Post for buying book | 3/4 | 3/4 | Shenyuan Done |
| Autocompletion for search | 3/4 | 3/4 | Hiroaki & Peiyan Done |
| User manual | 3/4 | 3/4 | Hiroaki Done |
| Iteration 1 presentation material | 3/4 | 3/4 | Hiroaki & all Done |
| **Iterations2 (Nice-to-have)** | 4/2 |  |  |
| Server Monitoring | 4/1 | 3/25 | Hiroaki Done |
| UI Improvement  - menu and transaction flow | 4/1 |  | Hiroaki |
| UI Improvement  - overall design | 4/1 |  | Amol |
| Category Detection | 4/1 | 3/30 | Hiroaki Done |
| Category Search | 4/1 | 4/1 | Shenyuan + Peiyan Done |
| Paypal Payment | 4/1 |  | Shaurya |
| Barcode Reader | 4/1 |  | Shenjie |
| Event Notification | 4/1 | 4/1 | Hiroaki Done |
| **Iteration3 (Nice-to-have)** | 4/30 |  |  |
| Remaining of iteration 2 | 4/29 |  | All |
| Integration Test | 4/29 |  | All |
| Return Book | 4/29 |  | Hiroaki + Shaurya |
| Auto Complete Book Cover | 4/29 |  | Shenyuan + Shenji |
| Recommendations engine | 4/29 |  | Amol + Peiyan |

\* See "Accomplishments Since Last Report" and "Plans to do for Next Report" for details of each item.

**Reference**

[1] https://aws.amazon.com/ec2/instance-types/

[2] https://www.x.org/releases/X11R7.7/doc/man/man1/Xvfb.1.xhtml

[3] https://www.cypress.io/

[4] https://www.npmjs.com/

[5] https://oss.oetiker.ch/mrtg/

[6] https://www.nginx.com/

[7] https://material-ui.com/getting-started/templates/dashboard/

[8] https://aws.amazon.com/education/awseducate/

[9] https://github.com/kamalkraj/BERT-SQuAD

[10] Jacob Devlin, Ming-Wei Chang, Kenton Lee, Kristina Toutanov, "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding"

[11] Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, Percy Liang. "Squad: 100,000+ questions formachine comprehension of text"

[12] https://aws.amazon.com/ebs/

[13] https://pytorch.org/

[14] https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/elastic-ip-addresses-eip.html

[15] https://www.paypal.me/

[16] https://www.dynamsoft.com/Products/Dynamic-Barcode-Reader.aspx

[17] Ian Sommerville, "ENGINEERING SOFTWARE PRODUCTS", p.273

[18] https://www.bu.edu/president/letter-to-the-community-on-coronavirus-followup/