

思い立った時に LT ができるプレゼンテーションツールの提案

辻永 泰輔 湯村 翼
(北海道情報大学)

1 はじめに

ライトニングトーク(Lightning Talk:LT) という, 5分程度の短い時間で発表するプレゼンテーション形式がある. IT 勉強会では, 開発の成果や悩みをテーマに LT を行う機会が多く存在する. LT を行うことで, 聴講者からのフィードバックを得て, 開発を前進させることができる.

LT の資料の作成には Microsoft PowerPoint や Google スライドといったプレゼンテーションツールを利用することが一般的である. これらのツールを用いた資料作成は, 質の高い資料が作成できるものの, 資料の作成に時間を要する. そのため, 開発を前進させたいという LT の動機に反して, 開発に利用できる時間を圧迫してしまう. LT の資料作成作業を省略することができれば, LT により開発を進める効果は高まる.

そこで本研究では, 資料作成作業を省略し, 思い立った時に LT ができるプレゼンテーションツール LTool(図 1)を提案する. LTool では, 開発中に記録した作業メモをもとに, 登壇中に都度スライドを作成する. 発表者が LTool に提示された選択肢を選ぶと, プレゼンテーション用スライドが生成され, スクリーンに表示される. このようにして, 発表者は事前の資料作成を行わずにプレゼンテーションを実施することができる.

2 LTool

2.1 作業メモ

LTool は, スライドのデータソースとして作業メモのデータを用いる. そのため, 開発作業時に作業メモを作成することを前提とする. LTool では, メモやノートを作成する Web サービス Scrapbox[1]を作業メモ用のツールとして利用する. Scrapbox は WebAPI を提供しているため, LTool は作業メモのデータを取得できる.

Scrapbox は, 1つのプロジェクト(図 2)の中に複数のページ(図 3)が存在する. 作業メモは Scrapbox ページ内に記述され, スライド作成に用いられる. LTool では, プレゼンテーション開始時に発表者がテーマを選択する. このテーマは Scrapbox のページに対応する.



図 1 LTool の概要

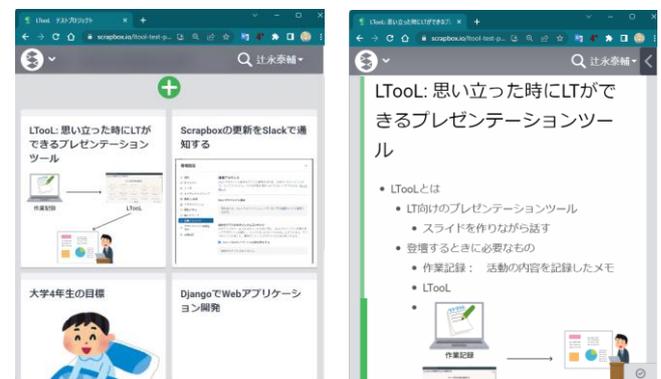


図 2 Scrapbox プロジェクト

図 3 Scrapbox ページ

2.2 プレゼンテーション

LTool のプレゼンテーションでは, 発表者は, 提示された選択肢のうち話したい内容を選択する. 内容選択後, LTool が生成するスライドを利用して発表する. これを繰り返すことで, プレゼンテーションを進行する.

まず初めに, 発表者はプレゼンテーションのテーマを決める. テーマ選択画面(図 4)の選択肢から, テーマとする内容をクリック操作で選択する. テーマを選択すると, テーマスライドが表示される (図 5).

次に, コンテンツ選択画面の選択肢から, 次に話すコンテンツをクリック操作で選択する (図 6). コンテンツを選択すると, コンテンツスライドが表示される (図 7). それ以降は, コンテンツの選択と表示をプレゼンテーション終了まで繰り返す.

2.3 質疑応答

LTool には質疑応答のための機能もある. コンテンツ選択画面(図 6)の選択肢のうち, 黄色で縁取りされたものが質問である. 質問を選択すると, 質疑応答スライド(図 8)が表示される.



図 4 テーマ選択画面



図 5 テーマスライド表示画面



図 6 コンテンツ選択画面

質疑応答スライドには、はじめは質問のみが記載される。発表者が口頭で回答すると、音声認識により回答内容がスライドに追記される。なお、質問は、Web フォームを利用し、聴講者からリアルタイムに収集する。この機能を質問箱と呼ぶ。また、プレゼンテーションを盛り上げるため、LTool が質問を生成する機能も盛り込む。質問生成には大規模言語モデル(LLM)を用い、作業メモのデータから予測できる質問を生成する。この機能をスピーカーインタビューと呼ぶ。

2.4 システム

LTool は Web アプリケーションとして実装する。Django で WebAPI を提供するサーバを実装し、ユーザーデータの管理を行う (図 9)。フロントエンドは Next.js で実装する。Scrapbox のデータは、非同期的に LTool サーバのデータベースへ保存する。プレゼンテーションの際には、ページの生成ごとに LTool サーバからデータを取得する。

本研究では、プロトタイプとして、フロントエンドのスライド表示機能のみを実装した。データは、プログラム内に変数として格納した。



図 7 コンテンツスライド表示画面

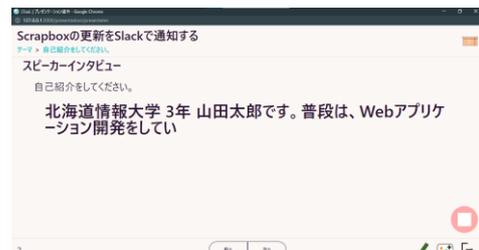


図 8 質疑応答スライド

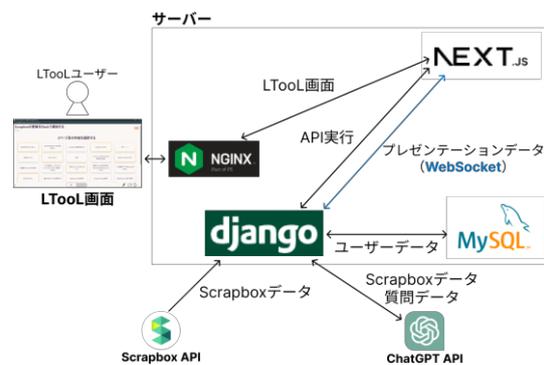


図 9 システム構成図

3 おわりに

本論文では、LTool の設計を示し、プロトタイプの実装を行った。今後、バックエンドも含めて LTool を実装するにあたり、課題のひとつが Scrapbox API の認証手続きである。Scrapbox プロジェクトのデータを取得する Scrapbox API へのアクセスは、scrapbox.io の Cookie に保存される認証情報が必要となる。Web ブラウザから認証情報を取得し、データベースに手動で登録する方法で実装可能であることを確認した。

LTool によって、LT 登壇の場を今まで以上に活かすことが可能になること、また、登壇へ動く人が増えることを期待する。

参考文献

- [1] Scrapbox, <https://scrapbox.io/>, 2016.