Dev-Sync

*Collaboration made easy.*

MAZIBUKO SP <208324071>

3/1/2018

Table of Contents

**Name of project**

**Domain analysis**

**General Field of business**

**Terminology**

* Customers and users
* Tasks and procedures currently performed
* Competing software
* Similarities to the domain

**Define the problem**

* Define problem domain
* Opportunity met by the system

**Define the scope**

**Vision and objective**

**Users of the system**

**Mandatory functions**

**Functional requirements**

**Requirements**

**Aspect of proposed system**

* Inputs and outputs should the system accept
* Computations the system should perform
* Timing and synchronization of events

**Non-functional requirements**

**Use case diagrams**

**Tools and technologies**

# **Name of project**

Dev-Sync System

# **Domain analysis**

### General field of Business

The project is aimed at providing a real time online help platform through join permissions for professionals, undergraduates and users alike to create ,update ,view and save documents, anytime and anywhere provided there is an internet connection.

### Terminology

User – A person who uses a system or software to achieve a particular goal

Owner – A user that creates file or document  
File - Storage unit for one or more documents.

Message - Written communication sent or left for a recipient.

Widget – Real-time file,

Gadget- API plug-in sign-in used either GitHub or FaceBook.

### Customers and users

Students

Interested users

Administrators

Developers

### Tasks and Procedures currently performed

### Competing Software

Some competing software’s include

ATOM - free and open source package created specifically for [Atom](https://atom.io/).

BRACKETS - This code editor by Adobe allows for the ability of real-time collaboration via an extension. ([http://brackets.io](http://brackets.io/))

CODESHARE - Codeshare is a web app service that requires code to be ported over to its service from your local environment,

### Similarities to the domain

Their platform allows you to collaborate on a number of different types of documents while also interacting with team members through their miniature social environment.

A video example of the system and additional information on other systems alike can be found in the link below:

<https://www.maketecheasier.com/5-free-alternatives-to-microsoft-sharepoint/>

https://webdesign.tutsplus.com/articles/real-time-code-collaboration-tools-for-developers--cms-30494

# **Define the problem**

### Define problem domain

Documents have become an important source of information all around the world .There are many platforms where documents can be shared, but only a few of these platforms offer collaborative editing in real time to documents. Many people who share documents and perform multiple editing to these documents often complain that they end up having many versions of the same document and not sure with the final version of a document.

Programmers often struggle to collaborate when writing code simultaneously therefore the ***Dev-Sync System*** is to fill that void by designing a web based application for collaborative document editing in real time, it is also important to obtain information about who ,how and when the document was last accessed

# **Define the scope**

The world is currently sitting at a point where instantaneous information is a vital key in the business factor, and life in general therefore creating a system that does not only allow individuals to share documents, but also enables them to edit the same document in real time, keep track of all the users of the system and possibly eliminating the issue of having multiple versions of the same document is the next step in the technology driven era.

The project seeks to provide an online platform with a potential target market of agencies and organizations that rely on virtual teams and any other individual who might have a need to collaborate real time information with someone located elsewhere.

The system should be capable of:

* Collecting and storing user information.
* Input and validation of data.
* Registering user and create a profile accounts for them.
* Users will have a screen that shows real-time edits made by invited users
* Viewing and editing other documents posted within the dashboard.
* Generating meta-data on files on the site.
* Sharing documents.
* Password protection.
* Organize and manage documents.
* Query online users.
* Send notification through email mentions to those who have access to a document.

# **Vision and objective**

The world is currently sitting at a point where instantaneous information is a vital key in the business factor, and life in general therefore creating a system that does not only allow individuals to share documents, but also enables them to edit the same document in real time, keep track of all the users of the system and possibly eliminating the issue of having multiple versions of the same document is the next step in the technology driven era.

The project seeks to provide an online platform with a potential target market of agencies and organizations that rely on virtual teams and any other individual who might have a need to collaborate real time information with someone located elsewhere.

# **Users of the system**

User – will operate the service as intended to collaborate work documents.

Administrator- In charge of maintaining the system and validating user information, the admin can update, add and remove information inserted by the user.

# **Mandatory functions**

* The system must allow user to register before they may login.
* The system must associate file correctly with users.

# **Functional requirements**

### Requirements

* All the user to register
* Allow the user/administrator to login and log out
* Allow users to type/post messages on the site
* Allow members to query for more details.
* Allow only registered users to update his/her personal details.

### Aspect of proposed system

* User will register their details by providing relevant information such as names and email address.
* To successfully login they will need to create a password.
* Once logged in, if the user wishes to create a document they are assigned as author therefore give permissions to other uses for editing created documents.
* Authors and only authors can delete documents created by them.
* User interested in created document can now edit document in real time.

### Inputs and outputs should the system accept

* API callbacks
* Service request
* File system (Input output)

### Computations the system should perform

* The system will generate metadata on the files
* The system will also handle disputes between entered strings of text thus insuring uniform streams.

Timing and synchronization of events

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Event** | **Trigger** | **Source** | **Use case** | **Response** | **Destination** |
| User creates account | Register details | User | Create Account | Successful – User created | Database |
| User login to system | Login in | User/Admin | Login | Successful login/login denied | Homepage |
| User Creates Profile | Create Profile | User | Create Profile | Successful –Profile created | Homepage |
| User creates document | Create document | User | Create File | New document created | Database |
| Invite a Collaborator | Add a contributor | Owner | Add Collaborator | Other users can are synched and can edit file | Database  Dashboard |
| Owner deletes document | Delete File | Owner | Delete File | Document deleted | Database |
| User query users | Post message | User | Query User | Response from other user | User |
| Edit profile | Change details | User | Edit Profile | Successful - Profile changed | Database |
| Change password | Forgot password | User | Change Password | Successful – Password changed | Homepage |
| Generate Report | Admin Login | Admin | Generate Report | Detailed report | Admin |

# **Non-functional requirements**

24 Hour accessibility

Allow administrator to add banner images (Customization)

Flexible service based architecture for future extension.

Ability to use FaceBook or GitHub Sign-ins, and to LIKE and SHARE the application with friends

# **Use case diagram**



# **Tools and technologies**

* Integrated Development Environment
  + - SUBLIME TEXT 3
    - NetBeans
    - NotePad++
* Frameworks
  + - Laravel 5.3
    - Vue.js
    - Node.js
* Languages (Scripting)
  + - PHP 5.6.4
    - JavaScript
    - CSS
    - HTML5
      * Twitter – Bootstrap Tech
* Composer Packages
  + - SocieBoy alerts
    - Gravatar
    - Laravel HTML collective
    - Tinker
    - Socialite
    - Symfony DOM-crawler
    - Intervention image
    - npm
* Application Programming Interface
  + - Google Drive Real-time
    - Google Developer - OAUTH 2.0
    - Facebook Developer –Sign In
    - GitHub Developer – Sign In
* Repository
  + - GitHub – Gits and Repo (Dev-sync)