Junior Project Proposal

**lorelines.com**

# Legal Notice

\_\_\_\_\_\_\_\_\_\_\_ makes no warranties of representation of any kind concerning the accuracy or suitability of the information contained in this document for any purpose. All such information is provided “as is” and with specific disclaimer of any warranties of merchantability, fitness for purpose, title and/or non-infringement. In no event shall \_\_\_\_\_\_\_\_\_\_\_\_ its employees or agents be liable for any direct, indirect or consequential damages resulting from the information provided in this document. This exclusion and limitation only applies to the contrary in any written license or subscription agreement from \_\_\_\_\_\_\_\_\_\_\_\_ in respect to this document.

# Copyright Notice

**Copyright © 2019-2020 \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ All Rights Reserved**

The copyright on all materials provided in this document in held by \_\_\_\_\_\_\_\_\_\_\_\_\_ or by the original creator of the material. Except as stated herein, none of the material may be copied, reproduced, distributed, republished, translated, posted, communicated to the public by telecommunications or transmitted in any dorm of by any means, including, but not limited to, electronic, mechanical, photocopying, recording, or otherwise, without prior written permission of the copyright holder. Permission is granted to display copy and distribute materials in this document for personal, non-commercial use provided no modifications are made to the materials and all copyright and other proprietary notices contained in the materials are retained. This permission terminates automatically if you breach any of these terms or conditions. Upon termination, any printed material must be immediately destroyed. Any unauthorized use of any material contained in this document may violate copyright laws, trademark laws, the laws of privacy and publicity, and communications regulations and statues. All rights, titles and interest not expressly granted are reserved.

# Revision History

|  |  |  |  |
| --- | --- | --- | --- |
| Author | Date | Version | Notes |
| Justin Boehnen | 10/19/19 | V0.1 | Added 4. Product Requirements |
| Isaac Medlin | 10/24/19 | V0.2 | Added 1. Introduction, Legal Notice and Copyright Notice |
| Seth Ray | 10/27/19 | V0.3 | Added 3. System General Description |
| Evan D. Clark | 10/27/19 | V0.4 | Added 2. Project Management |
| Justin Boehnen | 10/29/19 | V0.5 | Added 5. User Profiles |
| All | 10/29/19 | V 1.0 | Fixed formatting and made submission-ready |
| Seth Ray | 11/7/19 | V 1.1 | Fixed corrections |

# Signature Page

This document was accepted by:

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Signature Date

This document was submitted by:

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Signature of Justin Boehnen Date

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Signature of Evan Clark Date

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Signature of Seth Ray Date

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Signature of Isaac Medlin Date

Table of Contents

[Legal Notice i](#_Toc24047440)

[Copyright Notice i](#_Toc24047441)

[Revision History ii](#_Toc24047442)

[Signature Page iii](#_Toc24047443)

[1. Introduction 1](#_Toc24047444)

[1.1 Purpose 1](#_Toc24047445)

[1.2 Scope 1](#_Toc24047446)

[1.3 Intended Audience 1](#_Toc24047447)

[2. Project Management 1](#_Toc24047448)

[2.1 Change Management Procedure 1](#_Toc24047449)

[2.2 Software Delivery, Installation, and Acceptance Criteria 1](#_Toc24047450)

[2.3 Documentation and Online Help 1](#_Toc24047451)

[2.4 Project Risks 1](#_Toc24047452)

[2.5 Customer Responsibilities 1](#_Toc24047453)

[2.6 Status Reporting 2](#_Toc24047454)

[3. System General Description 2](#_Toc24047455)

[3.1 Problem Statement 2](#_Toc24047456)

[3.2 Major Subsystems 2](#_Toc24047457)

[3.3 Relation of System to Existing System(s) 2](#_Toc24047458)

[3.4 Hardware Platform Description 2](#_Toc24047459)

[3.5 Software Platform Description 2](#_Toc24047460)

[3.6 Third Party Libraries 2](#_Toc24047461)

[4. Product Requirements 3](#_Toc24047462)

[4.1 Terminology 3](#_Toc24047463)

[4.2 Functional 3](#_Toc24047464)

[4.3 Performance 5](#_Toc24047465)

[4.4 Reliability 5](#_Toc24047466)

[4.5 Data Description 5](#_Toc24047467)

[4.6 Security and Safety 5](#_Toc24047468)

[4.7 Constraints 5](#_Toc24047469)

[5. User Profiles 5](#_Toc24047470)

[Appendix A – Terminology 5](#_Toc24047471)

# 1. Introduction

## 1.1 Purpose

The purpose of this document is to propose the design and implementation of Lorelines. The sections here-after are dedicated to defining and describing the format of this document and the intended design plan. If this proposal is accepted, it will serve as a guide for the implementation process.

## 1.2 Scope

The scope of this document is limited to project management, general system discussion, and a description of the product requirements which will describe in limited detail the intended design features and functionality. Also, included is the specific functionality of modules and otherwise discrete functionalities that may later be added to the system.

## 1.3 Intended Audience

The intended audience is anyone who’s interested in designing their own story and any third party interested in this project.

# 2. Project Management

## 2.1 Change Management Procedure

In the event that there is a request to modify the project, the Change Administration Team (CAT) will be informed of the change via the Change Request Form found in Appendix A. The CAT currently includes all members of the project: Justin, Evan, Seth, and Isaac. Upon receiving a change request, the CAT will weigh the impact of the change on the project and either accept or decline the change. All requests will be compiled and saved in a text document on the project’s GitHub.

## 2.2 Software Delivery, Installation, and Acceptance Criteria

Lorelines will be completed and accessible via lorelines.com at the end of the 2020 Winter term. All documentation will be accessible from the project’s GitHub. Acceptance criteria will be based on the completeness and operability of the aforementioned items as determined by Todd Breedlove.

## 2.3 Documentation and Online Help

As stated in Section 2.2, documentation will be accessible through the project’s GitHub. Online help will be accessible through the completed Lorelines website.

## 2.4 Project Risks

Risks for this project include learning new technologies, balancing schedules for team meetings, and managing personal time to achieve project milestones.

## 2.5 Customer Responsibilities

The customer is expected to have access to the internet and an internet browser.

## 2.6 Status Reporting

Status reporting will be submitted weekly to Todd Breedlove. The report will include:

* Word completed during the past week
* Work to be completed during the next week
* Issues found during the previous week’s work
  + Potential way to fix the issues found

# 3. System General Description

## 3.1 Problem Statement

The purpose of this application is to help users create fictional stories and track their world building tools and documents. It will allow users to create custom entities that can contain info and data to help them keep track of their story. These custom entities will be able to be used in tandem with a timeline the user can create, to fully visualize and record events in the user’s story. Additionally, the application will have nested functionality with its custom entities, to allow the user ease of movement through the lore and background of their story. The user will be able to sign into their account from anywhere and have access to the lore and timeline for their story.

## 3.2 Major Subsystems

This system is comprised of two major subsystems, the database and the user interface.

The user interface will consist of a login screen which will greet every user. After logging in the user will see their available *lorelines*, and from there be able to access different areas, like a project in Visual Studio, or a Document in Microsoft Word.

The database will be used to hold all the users story information. It will hold the user’s login information, and all data contained in a single *loreline.* This includes all custom entities.

## 3.3 Relation of System to Existing System(s)

N/A

## 3.4 Hardware Platform Description

* Internet capabilities

## 3.5 Software Platform Description

Lorelines is designed to run in any browser that can support progressive web apps, and will be fully functional on mobile or desktop devices.

## 3.6 Third Party Libraries

The third-party libraries for this project will include language frameworks, such as: jQuery and React.

# 4. Product Requirements

## 4.1 Terminology

* *Custom entity:* a user-created or pre-built ‘class’ in which ideas, concepts, and user creations are stored and organized.
* *Entity color*: the color that instances of this entity will appear as on the *timeline*. The color will default to a random, bright color for users that do not wish to maintain a *timeline*.
* *Entity instance:* a custom entity with its fields populated with unique characteristic and identifiers.
* *Entity name*: the collection name for all instance of this entity.
* *Entity schema:* the layout of *fields* in a *custom entity.*
* *Field:* Text Field, Text Box, Check Box, Entity List and/or Image.
* *Instance link:* colored segments of text in *nodes* and *entity instances* that reference an *entity instance* and provide a link to view that instance.
* *Instance name:* the keyword that this instance is referred to by the timeline.
* *Link:* points on the left and right of *nodes* that represent ‘previous’ and ‘following’ events, visual *tethers* attach to these links.
* *Loreline:* The complete collection of *custom entities*, *entity instances*, and *timeline*, essentially a save file.
* *Node:* GUI objects that represent events and branching paths on the timeline.
* *Tether:* visual lines that connects nodes in a chronological sequence.
* *Timeline:* A record of events that consists of *nodes* and *tethers* that is created by the user.

## 4.2 Functional

1. Custom Entities
   1. Ideas are stored in instances of user-created classes and pre-built classes called *custom entities*
   2. Examples of pre-built *custom entities* include: Character, City and Event. These are entities that users are likely to use and demonstrate to users how to properly construct their own *custom entities*. Pre-built entities are an optional extra when creating a new *loreline*.
   3. *Custom Entity* Creator GUI
      1. Allows user to drag and drop *fields* onto an *entity schema*
      2. *Custom entities* can be saved, edited, deleted, and duplicated
      3. When creating a new *custom entity*, the user is required to define two variables: *entity name* and *entity color.* If the entity color is left blank, it will automatically choose one for the *entity.* When creating an *entity instance* only the *instance name* must be defined.
      4. Users will be prompted by a warning when attempting to delete an entity that has one or more *entity instances*
   4. How *entity instances* are referenced on the timelineand in *entity instances*
      1. When it is determined that the user has typed an *instance name* into a *node* or an *instance field* the *instance name* will be colored the *entity color*, creating an *instance link*
      2. When the user clicks on an *instance link* the referenced instance is shown.
2. Timeline
   1. The timeline will be on a grid that will allow easy placing of *nodes* and *tethers.*
   2. Users can place *nodes* on the timeline and connect them with *tethers* to other nodes.
      1. Nodes can automatically resize to fit the amount of text in them or be a set size with a scroll wheel.
      2. *Tethers* will automatically create a straight path from one *node* to another when dragged between the two.
   3. Users can use the *instance links* feature to minimize the amount of text on the timeline.
   4. User can export timeline as an image file (possibly a premium feature).
   5. Timelines will be automatically saved to avoid loss of data.
3. Hierarchical Directory
   1. A visual way to view custom entities and entity instances without use of the timeline. Visually similar to windows file explorer.
   2. Users will be able to search for and find custom entities that they have created using this system
4. User Accounts
   1. Creating user accounts
      1. User will be asked to provide the following information when creating their account:
         1. First name
         2. Last name
         3. Username
         4. Email address
         5. Birthday (For analytics)
         6. Country (For analytics)
         7. Password
            1. Encrypted password will be stored in database
5. Creating a new *loreline*
   1. Users can create as many *lorelines* as they like (likely a premium feature)
      1. Users will have the option of importing *entities* from previous *lorelines* for cases where a new story would require pre-existing material.
6. Stretch Goals
   1. Map Maker
      1. Users can create maps to be associated with certain entities, like a city or an island
      2. User can export map as an image file (likely a premium feature).
   2. Color Themes
      1. Light theme
      2. Dark theme
   3. Ads and revenue streams
      1. Optional one-time purchase for an ad-free experience and premium features

## 4.3 Performance

Lorelines will have slight load up times to due to accessing the database for user information. When the progressive web-app is loading, it will have loading screens so the consumer will know what’s happening.

## 4.4 Reliability

Lorelines should have a reliability of at least 95%. If we expect consumers to use the web-app, we need their information and storyboards as accessible as possible. Reliability is key for any storywriter that’s planning out a story.

## 4.5 Data Description

The data Lorelines will keep is each users’ custom *entities* and storyboards. Data will be kept from their first storyboards and entities to the one that their working on now. The data will primarily be text, pictures, and the storyboards themselves.

## 4.6 Security and Safety

Lorelines will hash and encrypt passwords while sending data to and from the database. Lorelines will also have to have additional security for payment if we reach our stretch goal of having ads and a premium service.

## 4.7 Constraints

Seeing as Lorelines is a sandbox storyboarding web-app, we don’t see it having to many constraints. Freedom of the consumer will be the key to the success for Lorelines.

# 5. User Profiles

The users of this web application will be people interested in storytelling, and sequence-building.

# Appendix A – Terminology

* *Custom entity:* a user-created or pre-built ‘class’ in which ideas, concepts, and user creations are stored and organized.
* *Entity color*: the color that instances of this entity will appear as on the *timeline*. The color will default to a random, bright color for users that do not wish to maintain a *timeline*.
* *Entity instance:* a custom entity with its fields populated with unique characteristic and identifiers.
* *Entity name*: the collection name for all instance of this entity.
* *Entity schema:* the layout of *fields* in a *custom entity.*
* *Field:* Text Field, Text Box, Check Box, Entity List and/or Image.
* *Instance link:* colored segments of text in *nodes* and *entity instances* that reference an *entity instance* and provide a link to view that instance.
* *Instance name:* the keyword that this instance is referred to by the timeline.
* *Link:* points on the left and right of *nodes* that represent ‘previous’ and ‘following’ events, visual *tethers* attach to these links.
* *Loreline:* The complete collection of *custom entities*, *entity instances*, and *timeline*, essentially a save file.
* *Node:* GUI objects that represent events and branching paths on the timeline.
* *Tether:* visual lines that connects nodes in a chronological sequence.
* *Timeline:* A record of events that consists of *nodes* and *tethers* that is created by the user.