Busy Bee

Version 1.0

Revision History

|  |  |  |  |
| --- | --- | --- | --- |
| **Date** | **Version** | **Description** | **Author** |
| 18/03/2020 | 1.0 | Document inception | Carla-Maria Rusu |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

Table of Contents

1. Introduction 4

1.1 Purpose 4

1.2 Scope 4

1.3 Definitions, Acronyms, and Abbreviations 4

1.4 References 4

1.5 Overview 4

2. Positioning 4

2.1 Problem Statement 4

2.2 Product Position Statement 4

3. Stakeholder and User Descriptions 5

3.1 Stakeholder Summary 5

3.2 User Summary 5

3.3 User Environment 6

4. Product Requirements 6

# Introduction

This document is meant to provide an overview of the high-level features and to describe a general functioning of the Busy Bee web application. The application is a user-friendly event-planner in which users can create and organize events and share them with other members. The technologies used are Spring Boot (backend), Vue.js (frontend) and MySQL (data storage). Explanations for acronyms and abbreviations, as well as references are provided in this document.

## Purpose

The purpose of this document is to collect, analyze, and define high-level needs and features of the event planning application, Busy Bee. It focuses on the capabilities needed by the stakeholders and the target users, and **why** these needs exist. The details of how Busy Bee fulfills these needs are detailed in the use-case and supplementary specifications.

## Scope

This document is associated with the event planning web application, Busy Bee. The application will need a working internet connection to function. It will be developed in Java, using Spring Boot for the backend and Vue.js for the frontend. A MySQL database will store the data. The application will provide a useful tool for the creation and management of events.

## Definitions, Acronyms, and Abbreviations

* Spring Boot is a Java-based framework used to create a micro Service. It is used to build stand-alone, production ready spring applications.
* Vue.js is a progressive framework for building user interfaces. It is designed to be incrementally adoptable, easy to pick-up and integrate with other libraries or existing projects.
* A complete list can be found in the Glossary

## References

* Project\_UseCaseModel
* Project\_SupplementarySpecification
* Project\_Glossary
* Bibliography

## Overview

The rest of this document contains the positioning, divided into problem statement and product position statement, the stakeholder and user descriptions and, finally, the product requirements.

# Positioning

## Problem Statement

|  |  |
| --- | --- |
| The problem of | Event planning and managing |
| affects | Individuals, groups, small businesses, families, etc. |
| the impact of which is | Disorganized planning, multiple to-do lists, missed deadlines |
| a successful solution would be | A user-friendly web application that can be accessed and modified by any member of an event in order to keep track of tasks, goals and deadlines |

## Product Position Statement

|  |  |
| --- | --- |
| For | Individuals, groups, small businesses, families |
| Who | Need to plan an event |
| The Busy Bee | Is a web application |
| That | Gives the users the ability to create and maintain a shared plan of an event, add tasks and goals and define deadlines |
| Unlike | Manually or mentally keeping track of lists and memos or orally planning an event |
| Our product | Safely stores the events in an organized way; is accessible anytime and anywhere; is intuitive and shared among members |

# Stakeholder and User Descriptions

## Stakeholder Summary

|  |  |  |
| --- | --- | --- |
| **Name** | **Description** | **Responsibilities** |
| Developer | The stakeholder responsible for the development of the application | ensures that the project will be operating  determines the design and implementation of the system  is responsible for the choice of technologies and architecture |

## User Summary

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Description** | **Responsibilities** | **Stakeholder** |
| Event creator | Primary end user of the system | Creates an event  Adds/removes members (optional)  Creates goals, tasks and deadlines  Determines the member’s permissions (view or edit) | Developer |
| Event member | End user of the system | Accepts/declines an invitation  Modifies goals, tasks, deadlines (if permission is granted)  Views goals, tasks, deadlines (if permission is restricted) | Developer |

## User Environment

* Number of users per event: at least one (the event creator); no upper limit is enforced; members can be added at any time.
* Data persistence: until a user with edit permissions deletes the event.
* Capabilities: registering, logging in/out, creating/deleting an event, adding tasks, goals, deadlines, checklists, members to an existing event.
* Environmental constraints: mobile, in-flight
* Platform: no support for mobile; works on any OS

# Product Requirements

The Busy Bee application requires a working Internet connection. The application is expected to be accessible to registered users. The data is expected to be stored correctly in an external database. The members’ permissions are to be enforced (no member with view permission should be able to edit).