GudSpot

**Software Development Plan (Small Project)**

**Version <1.0>**

**Revision History**

| **Date** | **Version** | **Description** | **Author** |
| --- | --- | --- | --- |
| <28/10/2021> | <1.0> | <Project Plan using given template> | <Khôi> |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

**Table of Contents**

[**1.**](#_147n2zr) **Introduction 4**

[*1.1*](#_3o7alnk) *Purpose 4*

[*1.2*](#_23ckvvd) *Scope 4*

[*1.3*](#_ihv636) *Overview 4*

[**2.**](#_32hioqz) **Project Overview 4**

[*2.1*](#_1hmsyys) *Project Purpose, Scope, and Objectives 4*

[*2.2*](#_41mghml) *Assumptions and Constraints 4*

[*2.3*](#_2grqrue) *Project Deliverables 5*

[**3.**](#_vx1227) **Project Organization 5**

[*3.1*](#_3fwokq0) *Organizational Structure 5*

[*3.2*](#_1v1yuxt) *Roles and Responsibilities 5*

[**4.**](#_4f1mdlm) **Management Process 5**

[*4.1*](#_2u6wntf) *Project Estimates 5*

[*4.2*](#_19c6y18) *Project Plan 5*

[4.2.1](#_3tbugp1) Phase Plan 5

[4.2.2](#_28h4qwu) Iteration Objectives 6

[4.2.3](#_nmf14n) Releases 6

[4.2.4](#_37m2jsg) Project Schedule 6

[4.2.5](#_1mrcu09) Project Resourcing 6

[*4.3*](#_46r0co2) *Project Monitoring and Control 6*

[4.3.1](#_2xcytpi) Requirements Management 6

[4.3.2](#_2lwamvv) Reporting and Measurement 7

[4.3.3](#_111kx3o) Risk Management 7

[4.3.4](#_49x2ik5) Configuration Management 7

**Software Development Plan (Small Project)**

# **Introduction**

The introduction of the **Software Development Plan** provides an overview of the entire document. It includes the purpose, scope, definitions, acronyms, abbreviations, references, and overview of this **Software Development Plan**.

## **Purpose**

The S*oftware Development Plan (SDN)* will define activities about developing GudSpot. It collects all information to describe the approach to the development of the project. All team members will use this SDN to focus on their workload in the future.

* The **team leader** uses it to plan the project schedule and resource needs, and to track progress against the schedule.
* **All team members** use it to understand what they need to do, when they need to do it, and what other activities they are dependent upon.

## **Scope**

This *Software Development Plan* describes the overall plan to be used by the GudSpot project, including deployment of the product. The plans as outlined in this document are based upon the product requirements as defined in the *Vision Document*.

## **Overview**

This *Software Development Plan* contains the following information:

Project Overview — provides a description of the project's purpose, scope, and objectives.  It also defines the deliverables that the project is expected to deliver.

Project Organization — describes the organizational structure of the project team.

# **Project Overview**

## **Project Purpose, Scope, and Objectives**

The goal of GudSpot is to create a website with a suggestion system that helps tourists when travelling. It also provides a community where tourists can share their experiences about places of interest.

The project also helps team members to learn how to follow software development processes to construct software.

## **Assumptions and Constraints**

The end of the semester will be a hard time since deadlines from other subjects will also be given.

Some team members are inexperienced in developing a software project and somes are not familiar with software development processes. As a result, this will limit our flexibility in the development process and the huge gap between team members' skill sets might delay project’s deployment.

Team members who enroll in many subjects this semester will also have many other projects and deadlines. For that reason, they will not be able to complete their tasks in reasonable time and the project will not be completed before the deadline.

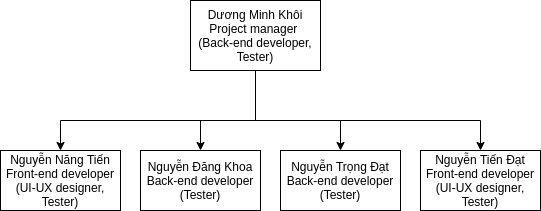
Because of the aboves reason, the project will not have a detailed, but large feature set.

## **Project Deliverables**

* Software Development Plan (this document)
* Vision Statement
* Use Case Model
* Use Case Specifications
* Use Case Realizations
* Development Case
* Software Architecture Document
* Iteration Plans
* Iteration Assessments
* Build
* Product: a website

# **Project Organization**

## **Organizational Structure**



## 

## **Roles and Responsibilities**

| **Person** | **Role** |
| --- | --- |
| Dương Minh Khôi: Project manager | Allocates resources, shapes priorities, tries to keep the project team focused on the right goal. The project manager establishes a set of practices to ensure the integrity and quality of project artifacts. |
| Nguyễn Trọng Đạt, Dương Minh Khôi, Nguyễn Đăng Khoa: Back-end developer | Design, create, control and maintain databases, API. Back-end team must ensure that communication between back-end and front-end will remain stable. During the development process, the back-end team should compose detailed documents for the front-end team, so that they can make good use of the back-end parts. |
| Nguyễn Năng Tiến, Nguyễn Tiến Đạt: Front-end developer | Front-end team is in charge of creating friendly UX and UI, creating responsive designs on different platforms, and maintaining project’s compatibility. |
| Dương Minh Khôi, Nguyễn Đăng Khoa, Nguyễn Trọng Đạt, Nguyễn Tiến Đạt, Nguyễn Năng Tiến: Tester | Review, analyze project development, prepare testing cases during the development process. Tester team is also responsible for executing test cases, analyzing problems, reporting bugs and errors for the development team, helping troubleshoot issues, and ensuring product’s quality. |

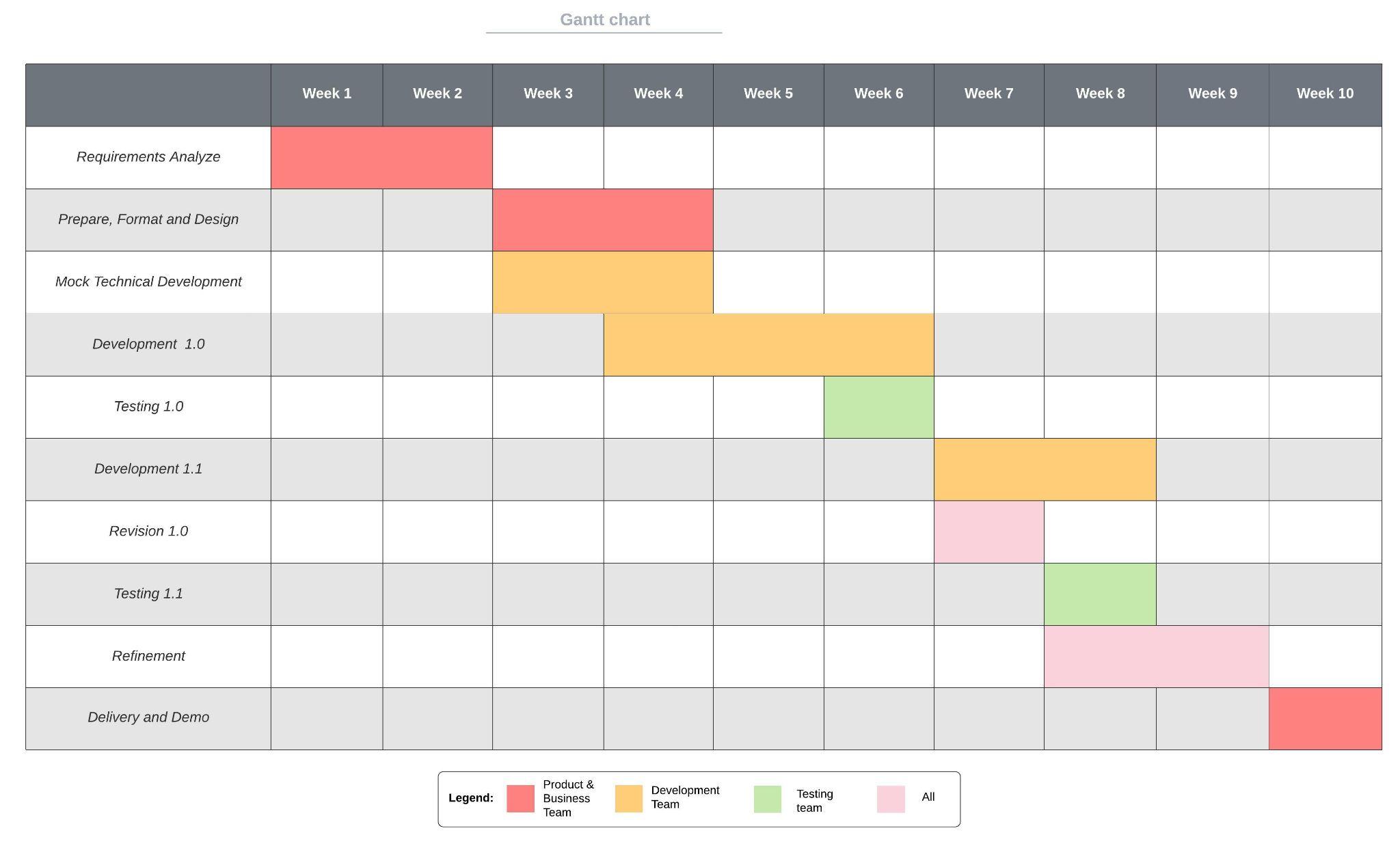
# **Management Process**

## **Project Estimates**

## **Project Plan**

### *Phase Plan*

| Phase | No. of Iterations | Start | End |
| --- | --- | --- | --- |
| Inception Phase | 1 | Week 1 | Week 2 |
| Elaboration Phase | 1 | Week 3 | Week 4 |
| Construction Phase | 2 | Week 5 | Week 10 |



### *Iteration Objectives*

| Phase | Iteration | Description | Associated Milestones | Risks Addressed |
| --- | --- | --- | --- | --- |
| Inception | Preliminary Iteration | Defines product requirements, project plan and cases. | Case Review | Determines feasibility of project from a business viewpoint.  Develops realistic project plans and scope. |
| Elaboration Phase | Develop Architectural Prototype | Completes analysis & design for all use cases. Develops the architectural prototype. | Architectural Prototype | Architectural issues clarified.  Technical risks mitigated.  Early prototype for user review. |
| Construction Phase | C1 Iteration – Develop Beta | Implement and test use cases to provide the Beta Version. | Beta | All key features from a user and architectural prospective implemented in the Beta. |
|  | C2 Iteration – Develop full Release | Incorporate enhancements and defects from initial release.  Develops the full system. | Software | Quick release addresses full requirements.  All key functionality provided in System by full Release. |

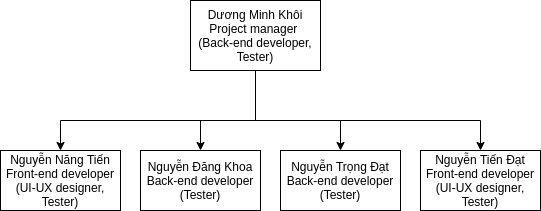
### *Releases*

The Beta version should build the skeleton functionalities for the product. The subsequent release will cover any remaining functionality and testing for Beta ver. The final release should completely satisfy the requirements with minimal abnormal cases.

### *Project Schedule*

| Version | Phase | Target Date |
| --- | --- | --- |
| Beta | Inception | October 17, 2021 |
| Beta | Elaboration | October 31, 2021 |
| 1.0 | Construction Iteration 1 | November 28, 2021 |
| 1.1 | Construction Iteration 2 | December 12, 2021 |

### *Project Resourcing*



* Resource Acquisition Plan : Not Applicable
* Training Plan: Most of the staff assigned to this project have appropriate skills at this point. A plan for skill adaptation will be updated in time to ensure the ability for the project.

## **Project Monitoring and Control**

### *Requirements Management*

The requirements for this system are captured in the Vision document. Requested changes to requirements are captured in Change Requests, and are approved as part of the Configuration Management process.

### *Reporting and Measurement*

Updated cost and schedule estimates, and metrics summary reports, will be generated at the end of each iteration.

The Minimal Set of Metrics, as described in the RUP [Guidelines: Metrics](about:blank), will be gathered on a weekly basis. These include:

Earned value for completed tasks. This is used to re-estimate the schedule and budget for the remainder of the project, and/or to identify need for scope changes.

Total defects open and closed – shown as a trend graph. This is used to help estimate the effort remaining to correct defects.

Acceptance test cases passing – shown as a trend graph. This is used to demonstrate progress to stakeholders.

In addition, overall costs will be monitored against the project budget.

### *Risk Management*

Risks will be identified in the Inception Phase using the steps identified in the RUP for Small Projects activity “Identify and Assess Risks”. Project risk is evaluated at least once per iteration and documented in this table. The risks of the greatest magnitude are listed first in the table.

| **Risk Ranking (High, Medium, Low)** | **Risk Description and Impact** | **Mitigation Strategy and/or Contingency Plan** |
| --- | --- | --- |
| High | Website fails to deploy within the established schedule  Project may be cancelled | The timeline must be considered carefully, should define all the core functionality in the first release. |
| High | Data and source go missing due to malfunctions of the system  Project may be cancelled | Prepare recovery method, store the content on more than one platform |
| Medium | Halting due to lacking of staffs ( drop out in the middle of progress or private issues)  Progress speed is degraded | Create a good environment to keep the staffs in a good condition  Have contingent substitution or outsource support |
| Low | Staff do not have appropriate skills for the project  Slow down the progress | Have proper acquisition or training. |

### *Configuration Management*

Appropriate tools will be selected which provide a database of Change Requests and a controlled versioned repository of project artifacts.

All source code, test scripts, and data files are included in baselines. Documentation related to the source code is also included in the baseline, such as design documentation. All customer deliverable artifacts are included in the final baseline of the iteration, including executables.