2023

5COM1079 – Software Development Exercise

Samuel Blinkhorne, Peter Feehan, Cameron Davidson

[Company name] | [Company address]

Contents

[Managing Software Development Projects 2](#_Toc138332024)

[A Comparison of Common Development Strategies (SB) 2](#_Toc138332025)

[Waterfall / V Methodology 2](#_Toc138332026)

[Agile 3](#_Toc138332027)

[Boehm Spiral 3](#_Toc138332028)

[Project Management Framework and Development Strategy (ALL/SB) 4](#_Toc138332029)

[Project Outline 5](#_Toc138332030)

[Problem Statement & Project Background (PF) 5](#_Toc138332031)

[Client Requirements (ALL/PF) 5](#_Toc138332032)

[Business Case - Project Goals & Benefits (ALL/PF) 5](#_Toc138332033)

[General Aims & Objectives (CD) 5](#_Toc138332034)

[Project Management and Development Sttegy 5](#_Toc138332035)

[Preferred Approach to Project Management and Development (SB) 5](#_Toc138332036)

[Provisional Project Gantt Chart and Task Schedule (SB) 5](#_Toc138332037)

[Project Risks (ALL / CD) 5](#_Toc138332038)

[Team Structure & Setup 5](#_Toc138332039)

[Team Members & Team Roles (CD) 5](#_Toc138332040)

[Software and Project Management Team Skills Matrix (CD) 5](#_Toc138332041)

[Development, Testing & Deployment 6](#_Toc138332042)

[Target Platform (PF) 6](#_Toc138332043)

[Development and Testing Platform (ALL / CD) 6](#_Toc138332044)

[Project Collaboration and Sharing (CD) 6](#_Toc138332045)

[Link to Online Repository (CD) 6](#_Toc138332046)

[System Requirements 6](#_Toc138332047)

[Existing System Use-Cases (ALL/CD) 6](#_Toc138332048)

[System Requirements 6](#_Toc138332049)

[Project’s Functional Requirements (ALL / PF) 6](#_Toc138332050)

[Project’s Non-Functional Requirements (ALL / CD) 6](#_Toc138332051)

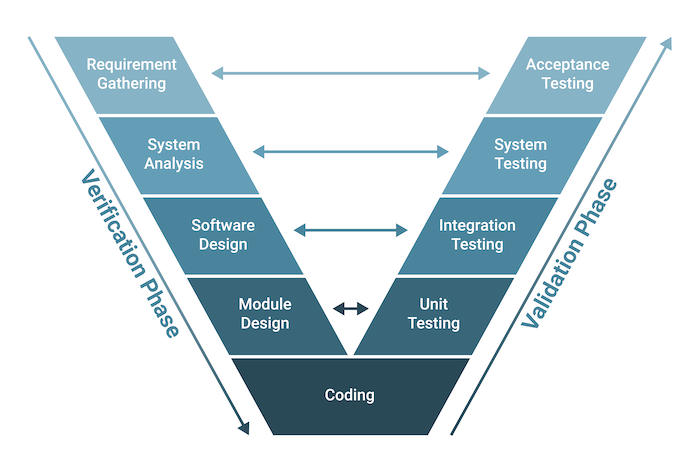
# Managing Software Development Projects

## A Comparison of Common Development Strategies (SB)

This section discusses three of the methodologies considered for the development of our software during this project and then states which development strategy is being used and justification for why this approach is the best for our team.

**Waterfall / V Methodology**

The waterfall method which can also be known as the ‘V’ method is one of the most established way of carrying out software engineering development. It was coined in the 1970s by Dr. Winston W. Royce (https://herts.instructure.com/courses/104211/files/7255858?module\_item\_id=3173003). It is a methodology that adopts a sequential approach to development meaning you are required to complete a stage of the development before moving on to the next stage. V methodology is an expansion on the standard Waterfall methodology that mitigates some of the disadvantages of the Waterfall methodology.



<https://builtin.com/software-engineering-perspectives/v-model>

Depending on verification phase you are in you take into consideration the opposite validation stage. This allows you come up with more effective requirements, design and analysis as you take into consideration how the system in the later stages will be tested at the various stages.

The Waterfall methodology is usually the most intensive when it comes to documentation. This can be an advantage for projects that have to be well documented or researched, such as in safety critical systems. But it can be a disadvantage for small team projects that don’t require a lot of documentation or for a project which needs to be very reactive to requirements changes.

Another advantage of this methodology is that everything has to be well established before moving onto the next stage this can be useful for developers as they have to have a full and complete design before they can begin development, so there is no ambiguity during development

### Agile

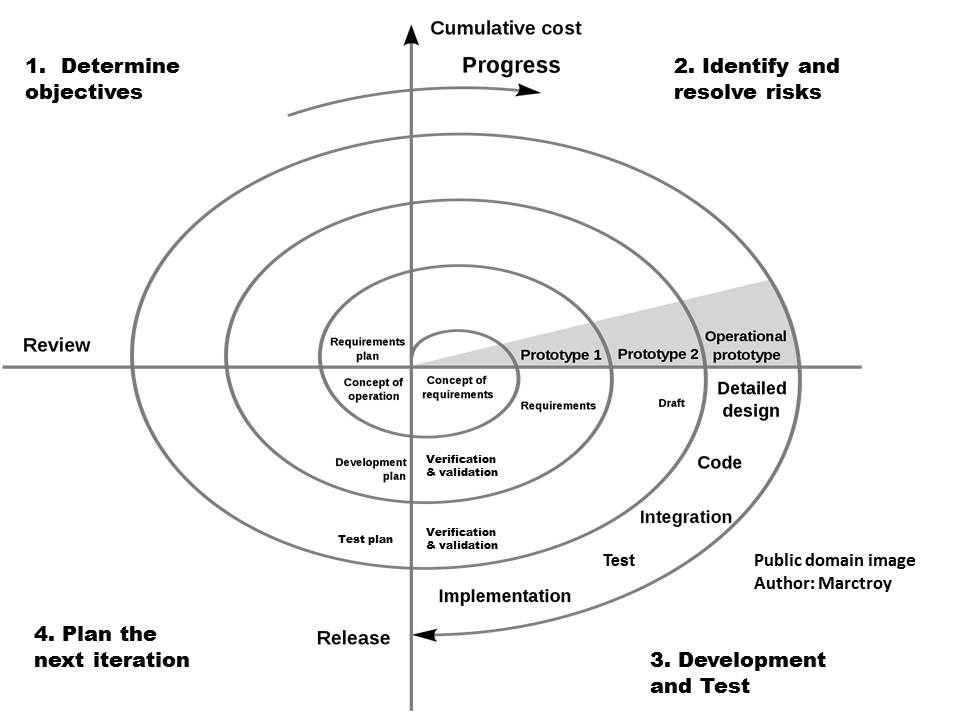
The Agile methodology was developed to make software development faster and more reactive requirements changes. It is a development strategy that shares similarities with iterative processes by takes it to a more extreme level. Agile has many aspects to it and can come in a variety of forms, including sprints, scrum meetings, use cases, Kanban and Extreme Programming (XP).

Agile methods rely on a lot of interaction with the customer which can add a lot of value to the development of the product as it allows for the customer to offer constant feedback on what they actually want as a product. The downside of this is if a customer is not responsive it can leave the team ‘in the dark’ with development and cause delays.

### Boehm Spiral

The Boehm Spiral is a type of iterative development strategy, developed by Barry W.Boemn (<https://ultimatesdlc.com/spiral-model/>). It is a model of development that works on carrying out 4 stages in an iterative way till the final product is developed. The four Stages are Determine Objectives, Identify and Resolve Risks, Development and Test and Plan The Next Iteration.

This development strategy allows for larger projects to be broken down into smaller chunks that can be defined and then developed 1 after the other. Allowing a smaller team to develop a bigger system without getting overwhelmed or having to define the entire system at the start of the project.

[https://ultimatesdlc.com/spiral-model/](https://ultimatesdlc.com/spiral-model/)

## Project Management Framework and Development Strategy (ALL/SB)

# Project Outline

### Problem Statement & Project Background (PF)

The client has asked us to create a new application that will forecast the weather for an area when provided data from the user. This differs from most well known and easily available weather predicting applications as it allows the user to input specific data about the current conditions where they are rather than relying on a black box system. Because this technology differs from the more commercially available products currently available the user experience must be intuitive and easy to use. That being said however it is likely that our target audience for this product will be looking for a more comprehensive system and with therefore be more open and willing to use a more complicated system for the pay off of more control. It has been requested that this project be accessible from a web application, this in addition to requiring a database of historic data means we must host the application on a web server. Finally we have been requested to allow logins for users so they may access this application. Because we also need to have administrators we must be careful to create a system that is both secure and keeps separation between different user privileges.

### Client Requirements (ALL/PF)

Definitions:

* Must:
* Should:
* Could:

Requirements:

1. Software must run as a web application
2. Software must allow users to login
3. Software must allow users to input current weather data
4. Software must present the user with a prediction after they input their data
5. Software must base predictions on past weather events
6. Software should be accessible from Android 12 and Windows 10 devices
7. Software should store historical data about past weather events
8. Only software admins should be able to create or edit historical data
9. Users should have an account before they are allowed to input data or be shown any predictions
10. Software could be accessible anywhere

### Business Case - Project Goals & Benefits (ALL/PF)

### General Aims & Objectives (CD)

# Project Management and Development Sttegy

### Preferred Approach to Project Management and Development (SB)

### Provisional Project Gantt Chart and Task Schedule (SB)

### Project Risks (ALL / CD)

# Team Structure & Setup

### Team Members & Team Roles (CD)

### Software and Project Management Team Skills Matrix (CD)

|  |  |  |  |
| --- | --- | --- | --- |
|  | SB | CD | PF |
| **Technical Writing** | 3 | 2 | 2 |
| **Software Development Techniques** | 3 | 1 | 3 |
| **Project Management** | 4 | 3 | 3 |
| **C# Software Language** | 3 | 0 | 0 |
| **Database Management** | 1 | 1 | 2 |
| **Database Building** | 1 | 1 | 1 |
| **HTML Language** | 1 | 3 | 1 |
| **Web Development** | 1 | 1 | 2 |
| **Server Development** | 2 | 1 | 2 |
| **Requirements Development** | 4 | 3 | 3 |
| **Jira** | 4 | 0 | 4 |
| **Github** | 1 | 1 | 1 |
| **Weather Algorithms** | 0 | 0 | 0 |
| **Weather Forecast Techniques** | 0 | 0 | 0 |
| **Design Document Writing** | 4 | 2 | 2 |
| **UI Design** | 3 | 4 | 3 |
| **UX Design** | 3 | 3 | 2 |

# Development, Testing & Deployment

### Target Platform (PF)

Our client requirements state that we are primarily developing a system accessible on both smartphone and PC. Therefore to effectively suit these needs we require a web-server as a primary platform. This way we can effectively deliver a system that works for both Windows 10 and 11 devices as well as latest Android versions. To effectively achieve this we plan to write our platform in c# as we have team members who are very capable with the language and its flexibility will allow for easy creation of a functional and effective web-server. Additionally for our web client we intend to use HTML5 alongside CSS as multiple team members are effective with it though have little experience so the large amount of documentation will assist us in creating an effective functional system.

### Development and Testing Platform (ALL / CD)

### Project Collaboration and Sharing (CD)

### Link to Online Repository (CD)

# System Requirements

### Existing System Use-Cases (ALL/CD)

### System Requirements

### Project’s Functional Requirements (ALL / PF)

* Users should be able to make an account using an id email and password
* Users should be able to login using their account
* Users should be able to input detail about the current weather conditions including:
  + location
  + pressure
  + humidity
  + temperature
* Admins should be able to input data into the database
* Admins should be able to reset users passwords
* Users shall be shown a prediction of weather after inputting data

### Project’s Non-Functional Requirements (ALL / CD)