Group 9: Sprint 1_ADR

1. Use of MsSQL over MongoDB for database

The group decided to use a persistent database for sprint 1. The host database is the mssql database which is hosted in the cloud by Microsoft Azure. The choice was because it has a student version with free 1Gb storage which should be enough for data storage purposes.

The website hosting service is also Microsoft Azure, so it made integrating the database easier as the group has access to the student subscription to MS Azure. This database uses SQL which is consistent with most databases. The alternative was MongoDB, but this is not linked to MS Azure.

Decision

We have decided to use MsSQL for our database.

Status

Accepted

Consequences

```
* SQL databases are in a tabular format, making it easy to read from and
write data to the database.
```

2. Managing project boards using GitHub instead of Trello.

Trello and GitHub provide a platform for programmers to manage their project, designate tasks, and track progress. Project boards help developers organize and prioritize tasks.

The project repository is created on GitHub for its development version control hence using its project board features would be easier than using Trello. With GitHub there is no need to manually update issues, bugs, or project statuses because the repository is linked. The assignment of tasks, pull request's or resolving conflicts is all executed with ease using GitHub over Trello.

Decision

We have decided to use GitHub to manage our project boards.

Status

Accepted

Consequences

```
* Updating the status of issues because it will automatically be done for
us.
```

3. Use of Trunk based development over GIT flow.

Trunk-based development works better for a schedule that has short iterations and more frequent releases. The short-lived feature branches allow each developer to work on a feature in isolation and then merge it into the main branch when the functionality has been achieved. The short-lived branches allow for small changes to be made during each development cycle. This allows for quick iterations and ensures that the product is in a functional, release ready state. Git flow has long lived branches that results in a longer period between each release and therefore less frequent releases.

Using trunk-based development allows for quicker iterations and more releases ensuring better feedback. The increased feedback results in a better product that satisfies the end user.

Decision

We have decided to use Trunk-based development.

Status

Accepted

Consequences

```
* Trunk based development allows each of us to work on a short-lived
feature branch and merge it into the main branch. We can keep each sprint
short and ensure that a working web application is deployed.
```

4. Use of RESTFUL Interface

We implemented a RESTFUL interface with routes for our web pages, this makes it simpler to define get and post requests for our webpages. This is in comparison to linking different html files in a single page.

Decision

We have decided to use a RESTFUL Interface.

Status

Accepted

Consequences

```
* RESTFUL interfaces made it easier to define routes for our web page which
made deploying it to AZURE easier.
```