Project Plan MDP

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Version** | **Date** | **Author(s)** | **Amendments** | **Status** |
| 1 | 27-11-2023 | Daniel Sales, Khaled Ali, Andrew Fleetwood-Bird, Timuçin Bulucu |  | Initial Draft |
| 2 | 1-12-2023 | Daniel Sales, Khaled Ali,  Andrew  Fleetwood-Bird, Timuçin Bulucu | Likelihood in the risk assessment, Mention the constraints of the project, Mention if the project is connected within constraints. | New amendments |
| 3 | 07-12-2023 | Daniel Sales, Khaled Ali,  Andrew  Fleetwood-Bird, Timuçin Bulucu | Describing the solution to the project objectives (the tasks steps, activities), the recalculation of the price for the project, sprints need to be added to the project plan (3 sprints, the goals of each sprint), | New amendments |
| 4 | - | Daniel Sales, Khaled Ali,  Andrew  Fleetwood-Bird, Timuçin Bulucu | - | - |
| 5 | - | Daniel Sales, Khaled Ali,  Andrew  Fleetwood-Bird, Timuçin Bulucu | - | - |

Contents

[Project Assignment 3](#_Toc152858597)

[1.1 **Goal of the project** 3](#_Toc152858598)

[1.2 **Context/Investigation** 3](#_Toc152858599)

[**2. Approach and Planning** 4](#_Toc152858600)

[2.1 **Approach** 4](#_Toc152858601)

[2.2 **Project Scope** 4](#_Toc152858602)

[2.2a **Project Objectives** 4](#_Toc152858603)

[2.2b Scope of Work: 5](#_Toc152858604)

[2.3 **Project Team** 5](#_Toc152858605)

[2.4 **Sprint deliverables** 6](#_Toc152858606)

[2.5 **Stakeholder** **Communication/Wishes** 7](#_Toc152858607)

[2.6 **Work Delivery** 7](#_Toc152858608)

[2.7 **Rough Product Breakdown** 7](#_Toc152858609)

[2.8 **Risk assessment** 8](#_Toc152858610)

[2.9 **Project Constraints** 9](#_Toc152858611)

# Project Assignment

# 1.1 **Goal of the project**

The purpose of this project is to increase website performance and availability by applying load balancing within a Kubernetes cluster.

Load balancing, a component of this entire infrastructure, is evenly dividing incoming network traffic across numerous cluster nodes. This technique results in accessibility, responsiveness, scalability, fault tolerance. Load balancing is an important aspect in sustaining smooth access, especially when user traffic increases.

The system effectively avoids bottlenecks by dynamically spreading incoming requests to multiple nodes, allowing the user to access the website's resources without suffering any delays or disruptions. Even during user spikes, Furthermore, scalability a component of the infrastructure is supported by the load balancing implementation inside a Kubernetes cluster.

The cluster will easily manage this growth by dynamically adding more nodes to the network as a website's popularity and user traffic rise. It is important to have dynamic scalability so that the website can adjust to changing requirements without sacrificing responsiveness or speed. Besides only being available and scalable, Load Balancing has other advantages. These have a significant impact on how quickly your page loads.

Overall, this solution enhances the user experience overall and decreases reaction time for users. Load balancing's benefits to fault tolerance, scalability, accessibility, and responsiveness.

# 1.2 **Context/Investigation**

Infrastructure students are going to support external clients and provide them customized services for their environment/product. This project revolves around collaboration, the groups must work together to achieve the highest quality of work for all parties involved. Every group will support 1 external client, the end services/product are not related/expected to be integrated with each other.

**External client**: Are formal clients in this project. They might develop their software and use the support from Infrastructure team and use the services provided by them

Since the client currently uses a load balancer, our objective is to duplicate its features to assess if it can increase cost effectiveness and efficiently handle the anticipated amount of traffic (1.5 million requests) that it receives.

# **2. Approach and Planning**

## 2.1 **Approach**

We will be using the Agile method to try to deliver something every two weeks. We will be using sprints to divide our work into parts. Stand-up meetings will be held at the beginning and at the end of each week, with also potential demos being shown-off. Meetings with the client will be done weekly in the beginning to keep them up to date with the progress of the project before switching to bi-weekly when necessary.

## 2.2 **Project Scope**

The Kubernetes Load Balancing project aims to enhance the performance and reliability of a web application by implementing and optimising load balancing strategies for network traffic between a web server and a database. The project will leverage Kubernetes' robust orchestration capabilities to efficiently distribute incoming requests across multiple nodes, ensuring optimal resource utilisation and high availability.

### 2.2a **Project Objectives**

1. Implement Kubernetes load balancing for web server and database nodes.
   1. Start in local environment using *minikube*, and then move to cloud provider (AWS)
   2. Test different configuration options in the *yml* files.
2. Optimise traffic management to evenly distribute loads and prevent bottlenecks.
   1. Investigate different methods for routing traffic between pods and the respective clusters.
   2. Create a load balancer that distributes the load across the pods evenly.
3. Enhance system scalability by dynamically adjusting to changing workloads.
   1. Setting accurate resource requests and limits within the Kubernetes environment to allow it to schedule and scale based on resource availability.
   2. Using the *HorizontalPodAutoscaler* to automatically adjust the number of replicas in the Deployment based on observed metrics.
4. Improve fault tolerance and reliability by minimising downtime during updates or failures. Monitor and analyse performance metrics to continuously improve load balancing strategies.
   1. Implementing strategies to minimize downtime during updates or failures in the environment.
   2. Setting up monitoring tools to collect performance metrics related to load balancing.

### 2.2b Scope of Work:

* **Kubernetes Cluster Setup:** Configure a Kubernetes cluster with nodes for the web server and database. Establish communication and networking between the nodes.
* **Load Balancer Implementation:** Select and configure a suitable Kubernetes load balancing solution. Define load balancing rules for distributing web and database traffic. Ensure proper integration with Kubernetes services and pods.
* **Traffic Management:** Auto-scaling based on demand dynamically adjusting the number of nodes/pods to handle levels of incoming traffic automatically.
* **High Availability and Fault Tolerance:** Set up redundancy and failover mechanisms to ensure continuous service availability.
* **Monitoring and Analysis:** Integrate monitoring tools to collect performance metrics.   
   Analyse data to identify performance bottlenecks and areas for optimization.  
  Implement alerts for abnormal behaviour or potential issues.

## 2.3 **Project Team**

|  |  |
| --- | --- |
| **Name** | **Role/tasks** |
| Daniel Sales  Khaled Ali  Timucin Bulucu  Andrew Fleetwood bird | Scrum Master/ Agile Team (Roosh)  Agile Team/Product Owner  Product Owner/ Agile Team (Fontys)  Agile Team |

## 2.4 **Sprint deliverables**

Sprint 1 (Week 11): Documentation

1: Research and Selection:

* + Research different load balancing solutions compatible with Kubernetes.
  + Evaluate pros and cons and select the most suitable option.

2: Documentation

* + Project goals and objectives.
  + Chosen load balancing solution and its features.
  + Proposed architecture within the Kubernetes cluster.
  + Steps and configurations required for implementation.
  + Testing and validation plans.

Sprint 2 (Week 15): Implementation, Delivery Tutor meeting Deliver URS, Design document & Test plan.

* Configure the cluster to support the load-balancing solution
* Implement the load-balancer within the Kubernetes environment
* Implement the ability to scale dynamically by adding nodes when needed
* Test said ability to scale dynamically

Sprint 3 (Week 17): Testing and Documentation Completion Test report, User & Technical manual, Process report & Demonstration

* + Test report
  + User technical manual
  + Process report
  + Documentation

## 2.5 **Stakeholder** **Communication/Wishes**

* Communication/meetings is through email/teams
* Planned for Thursdays/Friday’s
* Erik prefers mornings (in-person)
* Meetings are done weekly in the beginning of the project, after that switching to bi-weekly or every deliverable.
* Mikael prefers Thursdays in the afternoon

## 2.6 **Work Delivery**

After each sprint delivery, we share our findings and project approach with the client for approval. Once we're all on the same page about the discussed topics, we move on to the next phase of the project, continuing this cycle for each sprint. In the end, the completed work is bundled and submitted in a Gitlab repository.

## 2.7 **Rough Product Breakdown**

| **Component** | **Details** | **Estimated Monthly Cost** |
| --- | --- | --- |
| Number of EKS Clusters | 1 cluster | €144.00 |
| EC2 Instances | 5 t2.micro instances | €41.76 |
| Application Load Balancer (ALB) | Hourly Cost: €0.020 \* 730 hours | €14.60 |
| Worker node | 5 | €41.76 |
| **Total Estimated Monthly Cost** |  | **$283.88** |

## 2.8 **Risk assessment**

|  |
| --- |
| 1. Project team misunderstand requirements, even though we have tried to be as clear as possible within the description, it is possible that misinterpretation and misunderstanding of the project occurs.  Impact: **Catastrophic** (4) **Likelihood** 4/5 |
| **Solution**: Stand-ups, to prevent this, we have bi-weekly stand-ups of 15 minutes where the group discusses the current state of the project, while also reminding each other of the requirements and deadlines we must meet |
| 2. Clients have inaccurate expectations; the level of delivery could differ between project group and client. When miscommunication about the level of detail, or the actual product occurs. Impact: **Catastrophic** (4) **Likelihood** 3/5 |
| **Solution**: The client is kept up to date with communication about the project with also revisions being offered when the product differs from the wanted end-result |
| 3. Impacted individuals aren't kept informed, when the communication between the different parties is lost, participants don’t know where the current project is and can’t react properly to occurring issues.  Impact: **Catastrophic** (4) **Likelihood** 3/5 |
| Solution: Weekly meetings with group members and client, an overview of the project is created in the form of a Gann chart/Trello, so everyone can quickly see where it is and what has been finished |
| 4. Decisions are incomplete, when a discussion is had about a topic the decision must be adequate to continue the project, otherwise the project will get stuck quickly after.  Impact: **Medium** (2) **Likelihood** 3/5 |
| **Solution**: Discussions are kept to a minimal time limit, with the project lead having the final say about it to prevent bottlenecks. |
| 5. Project team lacks authority to complete work, when the project team is ill organised, and all work is subdivided with some of it missing. It can happen that some work is not delivered, however the whole project group is held accountable.  Impact: **High** (3) **Likelihood** 4/5 |
| **Solution**: Among the team there are “leaders” amongst tasks, where a person takes the lead on a certain task and the other person co-assists. Work is to be reviewed before submitted to prevent incomplete submissions |
| 6. Consequences, not everything always goes according to plan and there is a chance that certain promised features will not make it in the result due to time constraints or other reasons. This is something the project group is accountable for. Impact: **Catastrophic** (4) **Likelihood** 5/5 |
| **Solution**: Everyone is kept up to date with the project and plans are adjusted once time constraints or other factors are at play. It’s not always guaranteed that everything will go without a hitch, but we’ll adjust accordingly. |
| 7. When it comes to our project plan, we need to include things like addressing the content, problems investigation asking us what goal we are solving when it comes to coming up with a solution. **High** (3) Likelihood 3/5 |
| **Solution**:Be realistic, be specific, to know what problem you are trying to solve. Make sure that things aren’t vague. Try to figure out the bigger picture. |

## 2.9 **Project Constraints**

Due to the fact that we are students working on a company project for our school, our project has some special challenges. One major limitation is budgetary; rather than launching the project online, we're choosing to develop it locally due to financial constraints. This choice was made to control costs and prevent taking on more debt from server infrastructure and web hosting.  
  
The complexity is increased by the fact that we have a short deadline to complete our work and must balance our academic responsibilities with another ongoing project. Managing projects along with schoolwork is proving to be a difficult task. Since time is of the essence, agile development techniques, efficient task delegation, and careful planning are essential.