OSS Project Proposal

Team Name: Hungry Mile

Company Overview:

We are a small local online food ordering and food delivery platform operating in the Bay Area with 100 employees. We currently have an established food ordering app where customers can order meals from their favorite restaurants. Customers can choose to have their order delivered or they can pick it up themselves. Our user base in the Bay Area is growing and we are preparing to expand to other locations. We are also partnering with more restaurants. In addition to expanding our core business we are adding the capability to help restaurants reduce waste.

Food waste is a prevalent issue in the restaurant industry. In the United States, food waste is estimated at between 30-40 percent[1] of the food supply. Most wasted food ends up in landfills, where it generates methane, a greenhouse gas that is up to 86 times[2] more powerful than carbon dioxide. Even when someone is willing to donate the excess food, they do not know how to do that. To combat this problem, we want to partner with restaurants to offer extra food/ingredients to be sold at a discounted price rather than throwing away good food.

Current capabilities:

- Existing mobile app (Android & iOS)
- Existing database (Excel)
- Existing user base
- Existing restaurant partners
- Users can order food for pickup or delivery

Mission Statement:

We deliver your food and we go the extra mile for a better future.

Problem statement

At present the company is using Excel as a database due to which the app is suffering from

- Performance and scalability issues[3]: Our data is growing and it is becoming difficult to store and work with Excel.
- Version Control: It is becoming difficult to keep track of the versions of the Excel sheets. People have several copies of Excel and do not know which is the latest version.[3]
- Security issues: The data on Excel is also not secure even if it is password protected.[3]

Current Challenges:

1. Data Accuracy - Excel provides limited methods to validate input. This results in compromised and inaccurate data due to manual errors.[3]
2. Data Storage - Increased infrastructure cost with the onboarding of new restaurants and increasing customers.
3. Data Migration - Migration from current excel database to a larger database will be a time consuming process.
4. Data Security - Excel does not provide enhanced data security [3]. Data breaches affect the data integrity.
5. Data Recovery - Recovery and storage for failed scenarios is currently not available.

OSS Selected: *MySQL Database - https://www.mysql.com*

Reason:

1. Easier to manage large datasets: Large datasets slow down the computer when accessed through Excel while MySQL makes managing large datasets easier and does not slow down the system.[4]
2. Relational Database: Excel is a flat file database while MySQL is a relational database which makes accessing data/tables easier.
3. Scalability: Using Excel increases the server response time and creates issues with scalability while MySQL will help overcome scalability issues.[5]
4. Data Security: MySQL provides end-to-end data security and is compliant with the major privacy regulations and data protection laws.[5][6]
5. Improved Page Load time: Page Load Time and Page Response Time is an important aspect of customer experience and with a growing database, Excel would slow down the response time while MySQL provides improved application performance.[5]
6. Backup and Recovery: MySQL offers various methods for creating backups.[7]

Hypothesis :

When working with large databases that require speed, accuracy, consistency, volume, and power, MySQL has proven to be better than Excel [8]. The data stored in MySQL is easier to manage (manipulate/ filter/ update/ integrate), can be efficiently accessed (retrieved/ stored) and more secure than that in Excel as the data grows.

Success Criteria and KPIs:

1. Cost: Calculate the cost of redesigning and migrating to the new infrastructure.
2. Response Time: Improved response time for queries and improved Page Load and Response Time would indicate a successful database migration.[9]
3. Latency Errors: Latency gauges the time it takes an application to respond to a request.[10]

4. Query Throughput: This is the speed at which the database server is capable of implementing store/update/delete/retrieve queries.[9]
5. Open Connections: Database connections are network connections that facilitate communication between client software and database software. Open connections are necessary for sending commands and receiving responses in result sets.[9]

References:

1. “Food Waste FAQs”, USDA.gov(https://www.usda.gov/foodwaste/faqs)
2. McDonald, J. (2018) “How Potent is Methane?”, Factcheck.com(https://www.factcheck.org/2018/09/how-potent-is-methane/)
3. Cacace, M. (2014) “7 Reasons to Beware of Using Excel as a Database”, Bound States Software (https://boundstatesoftware.com/blog/7-reasons-to-beware-of-using-excel-as-a-database)
4. Hershy, A. (2019), “Excel v/s MySQL: A Conceptual Comparison”, Towards Data Science (https://towardsdatascience.com/excel-vs-sql-a-conceptual-comparison-dcfbee640c83)
5. “Custella Boosts its SaaS Application”, MySQL (https://www.mysql.com/why-mysql/case-studies/custella-saas-migrating-to-mysql-database-service.html)
6. “Chapter 6 Security”, MySQL(https://dev.mysql.com/doc/refman/8.0/en/security.html)
7. “Chapter 7 Backup and Recovery”, MySQL(https://dev.mysql.com/doc/refman/8.0/en/backup-and-recovery.html)
8. MOOC Blog Team, NOV 23, 2021, “Excel vs SQL” (https://www.mooc.org/blog/excel-vs-sql)
9. (2021), “8 Crucial Database Performance Metrics”, ScoutAPM (https://scoutapm.com/blog/database-performance-metrics)
10. (2021), “Migrating to the Cloud? Here are 7 metrics to consider”, Own Backup (https://www.ownbackup.com/blog/cloud-migration-metrics/)