**Online Cake Delivery**

**Team Name:**

SAAS

**Team Members**

Anvesh Athmakuri 11116468

Ashik Shaik 11167689

Srikanth Babu Pusapati 11157128

Sudhira Badugu 11157939

1. **Project Description**

Lives are getting Hectic day by day and buying cakes in these busy schedules is bit tough. To make it simple and also to deliver the cakes through the available people that voluntarily registers online; making this a new value added feature to the customer, for a faster and user friendly approach. The main objective of this application is to deliver cakes to any corner of the city using the interactive website.

This project will help in revolutionize the way events, parties are celebrated across the city, with no permanent staff and scheduling issues, delivery on time will be taken care by using a single site with a few clicks of buttons.

1. **Functional Modules:**
2. User module: User registers online with contact details through the site and logs in to view the available cakes with total costs involved. User can request the specific time to deliver and place of his choice.
3. Cake Deliverer: The worker registers by giving his credentials and logs in to view the available orders to be delivered including the details to reach out. And can accept any of the order according to his availability and his details of contact will be send to the customer.
4. Admin: Admin has his own login credentials and more control over the website. He performs actions like updating cake availability in the website, validating the delivery schedules etc.
5. **Implementation Details**
6. Tools:

* XAMP controller – For establishing connection to various ports in local system.
* Tortoise Git – version control system.

Backend:

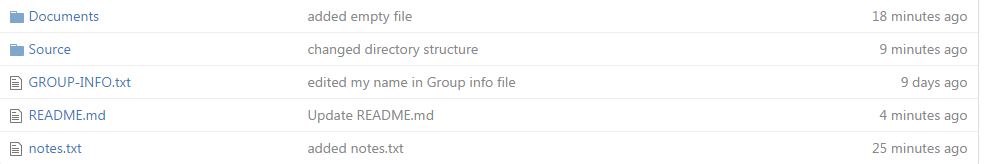
* MySQL distributed using php myadmin.

Frontend:

* PHP v5.4 – For server side scripting.
* Html 5 – Basic look and feel of the website UI.
* CSS3 – Adding other style features for the website UI.

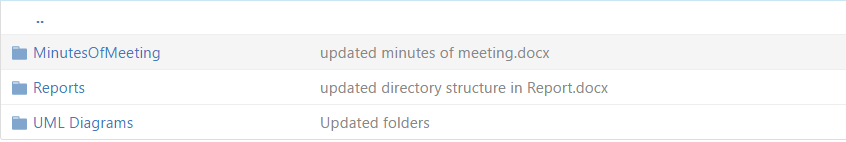
1. **Project Directory Structure**

Major directories in project

****

1. **Documents:**

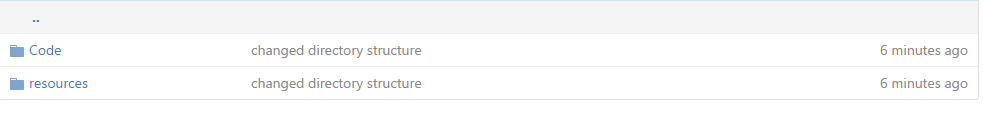
The document directory structure contains the following sub documents which is used to various documents.

****

* 1. Minutes of Meeting**:** This directory consists of all the scheduled meeting documents.
  2. Reports**:** Project documentation and deliverable report.
  3. UML Diagrams**:** All the diagrams to be drawn would be in this folder.

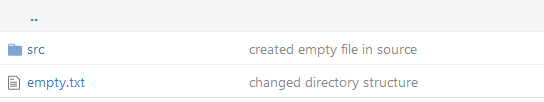
1. **Source:**

The source directory consists of all the files related to code.



Source contains two folders:

1. Code: Code consists of all the implementation files.



1. Resources: This folder consist of images, libraries etc.
2. **Minutes of meeting**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| DATE | DISCUSSION HIGHLIGHTS | TIME | MEMEMBERS ATTENDED | ATTENDEES |
| 30 TH AUG | Project idea brain storming | 1hr 30min | 4 | Ashik, Anvesh, Srikanth, Sudhira |
| 1 ST SEP | Defining Functionalities precisely | 2 hours | 4 | Ashik, Anvesh, Srikanth, Sudhira |
| 4 TH SEP | Research on platforms and languages | 1hr | 4 | Ashik, Anvesh, Srikanth, Sudhira |
| 6TH SEP | Identifying the risks in the project | 1hr. 20 min | 4 | Ashik, Anvesh, Srikanth, Sudhira |
| 8TH SEP | Gathering the requirements and installations | 1hr. | 4 | Ashik, Anvesh, Srikanth, Sudhira |
| 9TH SEP | Work Sharing of Delivarble -1 | 3hrs | 4 | Ashik, Anvesh, Srikanth, Sudhira |
| 10TH SEP | Presentation of Delivearble -1 | 1hr | 4 | Ashik, Anvesh, Srikanth, Sudhira |
| 11TH SEP | Gantt Chart making | 1hr | 4 | Ashik, Anvesh, Srikanth, Sudhira |
| 13 TH SEP | Study and quick review of concepts | 50 mins | 4 | Ashik, Anvesh, Srikanth, Sudhira |

**Meeting 1**

**Date:** 30th August

## **Points Discussed**

1. The major purpose of the project.
2. Defining the major actors and interaction between them.
3. Visualizing the project.

**Meeting 2**

**Date:** 1st September

## **Points Discussed**

1. Requirement gathering.
2. Project functionalities defined.
3. Researched on the project required languages, frameworks that can be leveraged.

**Meeting 3**

**Date:** 4th September

## **Points Discussed**

1. Discussed on the various available platforms to work on project.
2. Interaction between team members they were comfortable to code.
3. Finding strengths of each member of team.
4. Finalized languages to be used

**Meeting 4**

**Date:** 6th September

## **Points Discussed**

1. Identifying the project deadlines for completion.
2. Making a vague idea of the risks may occur.
3. Classifying and listing the risks and making the risk management table.

**Meeting 5**

**Date:** 8th September

## **Points Discussed**

1. Installing and getting versed with the push, pull functions.
2. Getting the required software installed like XAMPP server.
3. Making sure everyone is flexible with actions can be performed.

**Meeting 6**

**Date:** 9th September

## **Points Discussed**

1. Sharing the work mentioned in Deliverable-I
2. Quick view of the functionalities and problems may occur during implementation.

**Meeting 7**

**Date:** 10th September

## **Points Discussed**

1. Dividing the modules to be covered in the presentation.
2. Making the presentation.

**Meeting 8**

**Date**: 11th September

## **Points Discussed**

1. Making a review on the concepts required.
2. Discussion of implementing methods.
3. **Gantt Chart**

Gantt charts illustrate the start and finish dates of the terminal elements and summary elements of a project. Terminal elements and summary elements comprise the work break down structure of the project

**1. Clarity**

One of the biggest benefits of Gantt chart is the tool's ability to boil down multiple tasks and timelines into a single document. Stakeholders throughout an organization can easily understand where teams are in a process while grasping the ways in which independent elements come together toward project completion.

**2. Communication**

Teams can use Gantt charts to replace meetings and enhance other status updates. Simply clarifying chart positions offers an easy, visual method to help team members understand task progress.

**3. Motivation**

Some teams or team members become more effective when faced with a form of external motivation. Gantt charts offer teams the ability to focus work at the front of a task timeline, or at the tail end of a chart segment. Both types of team members can find Gantt charts meaningful as they plug their own work habits into the overall project schedule.

**4. Coordination**

For project managers and resource schedulers, the benefits of a Gantt chart include the ability to sequence events and reduce the potential for overburdening team members. Some project managers even use combinations of charts to break down projects into more manageable sets of tasks.

**5. Creativity**

Sometimes, a lack of time or resources forces project managers and teams to find creative solutions. Seeing how individual tasks intertwine on Gantt charts often encourages new partnerships and collaborations that might not have evolved under traditional task assignment systems.

We have drawn this chart with given values in the below table that describes the high level planning of project using Excel sheet. The sheet has also been included in the directory.

**Project Planning Table:**



1. **Risk Management**

Risk management refers to the practice of identifying potential risks in advance, analysing them and taking precautionary steps to reduce/curb the risk. The following table includes the various risks and their priorities and its mitigation plans. It also includes the Re-Evaluation plan as the time progresses.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| S.No | Description | Comments | Contingency plans | Re-Evaluation |
|  | Requirement gathering | Need to freeze on the requirements within scheduled time line. | Need to finalize on the project requirements specifications. | Identify all the project requirements are noted down, if modified re-evaluate the scope and project timeline. |
|  | Learning curve | All of the team members need to be in par with the development software’s, languages being used within the project. | Need to be aware of the programing languages being used like Java, PHP and tools like xampp controller. | Ensure that each component is well understood, visualized and the team have enough knowledge to implement it on time. |
|  | Ensuring Quality | Need to evaluate the testing strategy for the project. | Ensure the testing scenarios are updated in par with the development life cycle. | Ensure Writing test cases are done in parallel to the development. |
|  | Ensuring Quality | Need to use high quality images | Ensure Customer is satisfied with the UI | Should use high quality HDPI/XHDPI images during the design and development. |
|  | Customer Satisfaction | Need to ensure the product ordered is available in time for delivery. | Ensure customer friendly website with delivery in time. | Customer should be satisfied with the services provided |

1. **Project Report**

|  |  |  |  |
| --- | --- | --- | --- |
| **S.no** | **Name** | **Contribution** | **Contribution**  **Description** |
| 1 | Anvesh Athmakuri | 25% | Core Idea of the project, specified requirements and Filling the gaps between Team members |
| 2 | Ashik Shaik | 25% | Research on various platforms that suits best and visualization of working of each module. Gantt Chart preparation. |
| 3 | Srikanth Pusapati | 25% | Flexible with GitHub, risk management and Backend implementation issues. |
| 4 | Sudhira Badugu | 25% | Precise view on each module functioning and Documenting it, scheduling meetings.  Project director structure readme |

With these roles established, the project repository checkout and update policies that we are trying to implement in the project are:

1. Every commit should get pushed into the master branch, to reduce different team members to be in different branches other than the main branch.
2. Every commit prior to the push into the repository should do a pull to check if any conflicts occurs, which reduces code rework or merge conflicts with other team members.
3. Every commit should ensure that the code doesn’t breakdown during deployment in other team members system.

Central Repository  
Master Branch

Commit, working code.

Check-out, resolve conflicts

Developer system

**References:**

<https://en.wikipedia.org/wiki/Gantt_chart>

<https://www.youtube.com/watch?v=TjxL_hQn5w0>

<http://economictimes.indiatimes.com/definition/risk-management>