PIZZA CRUSH

Group: 8

Team Members:

1. Sourab Reddy Pailla (Group Leader)
2. Gowtham Kesa
3. Rishi Reddy Kolanu
4. Nagendra Beesabathuni

Project Description:
Pizza Crush is a web-based application pizza ordering system which allows customer to either order pizzas from menu or order their favorite customized pizzas by selecting the required crust, toppings and sauce.

Features:

- Ordering pizzas from the menu as in existing system.
- Ordering their customized pizzas by selecting breads, toppings, sauces.
- The bill is generated dynamically and is displayed as soon as customer selects an item.
- Customer can view the final image after selecting custom pizza and toppings. This image is generated by overlapping pizza and toppings images.

Nice to Have:

- Payment Gateway
- Email confirmation of orders

Limitations:

- Deployment – Local

Technologies used / Technical requirements:

- CodeIgniter (PHP framework)
- JavaScript
- CSS
- HTML5
- MYSQL
- Bootstrap CSS
- Apache Tomcat Server
- Wamp Server

Hardware requirements for hosting the website locally:

Processor: Intel Core i3 or above

Processor speed: 2.4 GHz or above

RAM: 4 Gb or more

Hard disk: 500 Gb or more

Non-Technical requirements:

1. Reliability: Reliability is the probability that a system will fail in a given period. A distributed system is reliable if it keeps delivering its service even when one or multiple components fail. Reliability is achieved through redundancy of components and data (remove every single point of failure).

2. Availability: Availability is the time a system remains operational to perform its required function in a specific period. Measured by the percentage of time that a system remains operational under normal conditions.

3. Efficiency:

   Latency: response time, the delay to obtain the first piece of data.

   Bandwidth: throughput, amount of data delivered in a given time.

Project Management tool: Tiago.io
Taiga is a project management platform for agile developers & designers who want a simple, beautiful tool that makes work truly enjoyable.

Version Control System: Git hub
GitHub is a Git repository hosting service. It is a repo for the code base of the project.

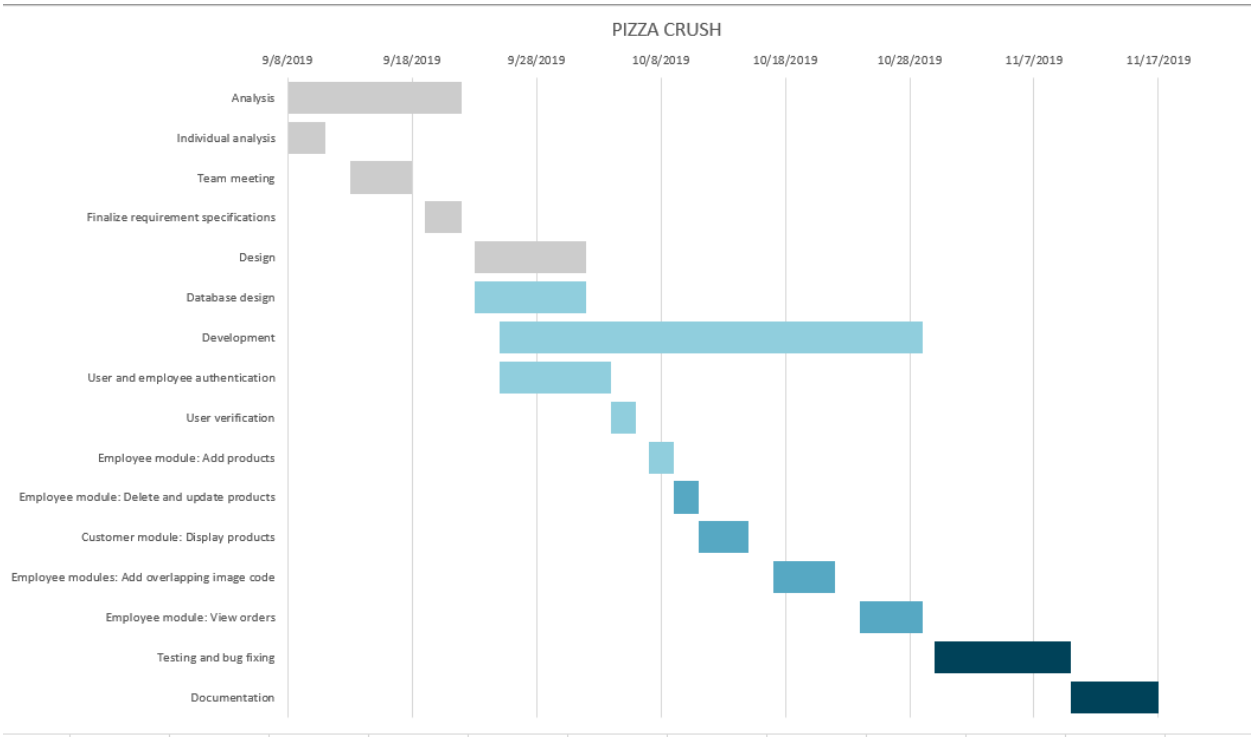
Development Methodology: Agile
Agile development methodology is being used to minimize risk (such as bugs, cost overruns, and changing requirements) when adding new functionality. Team develop the software in iterations that contain mini-increments of the new functionality that is being considered.

Plan for Structure of Project:

```
|__application
     |_cache
     |_ci_user
     |_config
     |_controllers
     |_core
     |_errors
     |_models
     |_views
     |_controllers
     |_libraries
|__assets
    |_images
       |_breads
```

|_extras
|_sauce
|_toppings
|__system
|__user_guide
|__ .htaccess
|__ index.php

Gantt chart:



Gantt chart for the project

Risk Management:

1. *Security risk*:
   Analysis - For every project threat modelling is an important stage. Threat modeling is a family of activities for improving security by identifying objectives and vulnerabilities, and then defining countermeasures to prevent, or mitigate the effects of, threats to the system. A threat is a potential or actual undesirable event that may be malicious (such as DoS attack) or incidental (failure of a Storage Device). Threat modeling is a planned activity for identifying and assessing application threats and vulnerabilities. It might become difficult to address security related issues are we lack the expert in this region and due to time constraints as well we consider this as a major risk. To mitigate this risk security expert is identified and it is timely monitored.

2. *Testing and Test scripts*: Implementing risk analysis in software testing typically requires a detailed evaluation of the source code to identify how it interacts with other components of a complete application. This evaluation looks at the various code components and maps how the code interacts. With this map, transactions can be identified and evaluated. Architectural and structural rules can be applied to the map to understand where software flaws lie, and which ones are the most important given the transactions flowing through the application. Such kind of risk might rise due to lack of knowledge on the testing scripts. To mitigate this risk testing expert is ramping up on mastering the testing and writing scripts.

3. *UX Design:* Most new products are probably carefully designed around user needs, user stories or typical personas. However, this is not always going to guarantee that your product will be useful and desirable or that it will be a commercial success. So, there is always a need for a UX advocate to tackle the failures of the product caused due to bad UI design. To mitigate such kind of a risk a UX advocate is identified and the tasks are monitored timely to ensure the design is simple to use for the user.

Team members roles and responsibilities:

1. Sourab Reddy Pailla (Group Leader and Manager): As a group leader he is responsible to resolve all the hurdles the team is facing. Team unity must be ensured. He is been identified as the UX advocate contact and is responsible for finding out the best UI design for the application and create the mockups for the application. Also, everyone is a developer at the end of the day and accountable for the 25% of the coding part.

2. Gowtham Kesa (Product owner and Security expert): As a product owner he is responsible to analyze the requirements of the product and defining the scope of the project according to the deadlines. Need to ensure the development of the project is done according to the requirement. As a security expert, threat modelling has to be performed on the product once it is ready and accountable for the 25% of the coding part.

3. Rishi Reddy Kolanu (Architect and Developer): As an architect he is responsible for designing the whole architecture of the product. All the problems and difficulties in each and every phase of development and testing are to be resolved by the architect and needs to ensure the development is being done according to the architecture that he/she has defined. As a developer he is accountable for the 25% of coding part.

4. Nagendra (Scrum Master and Testing Lead): As a scrum master he is responsible for undertaking all scrum activities and as a testing lead, he is responsible for leading the testing phase and writing the scripts.