# Ideation Phase Brainstorm & Idea Prioritization Template

|  |  |
| --- | --- |
| Date | 25 June 2025 |
| Team ID | LTVIP2025TMID33542 |
| Project Name | Grainpalette-A deep learning odyssey in rice type classification through |
| Maximum Marks | 4 Marks |

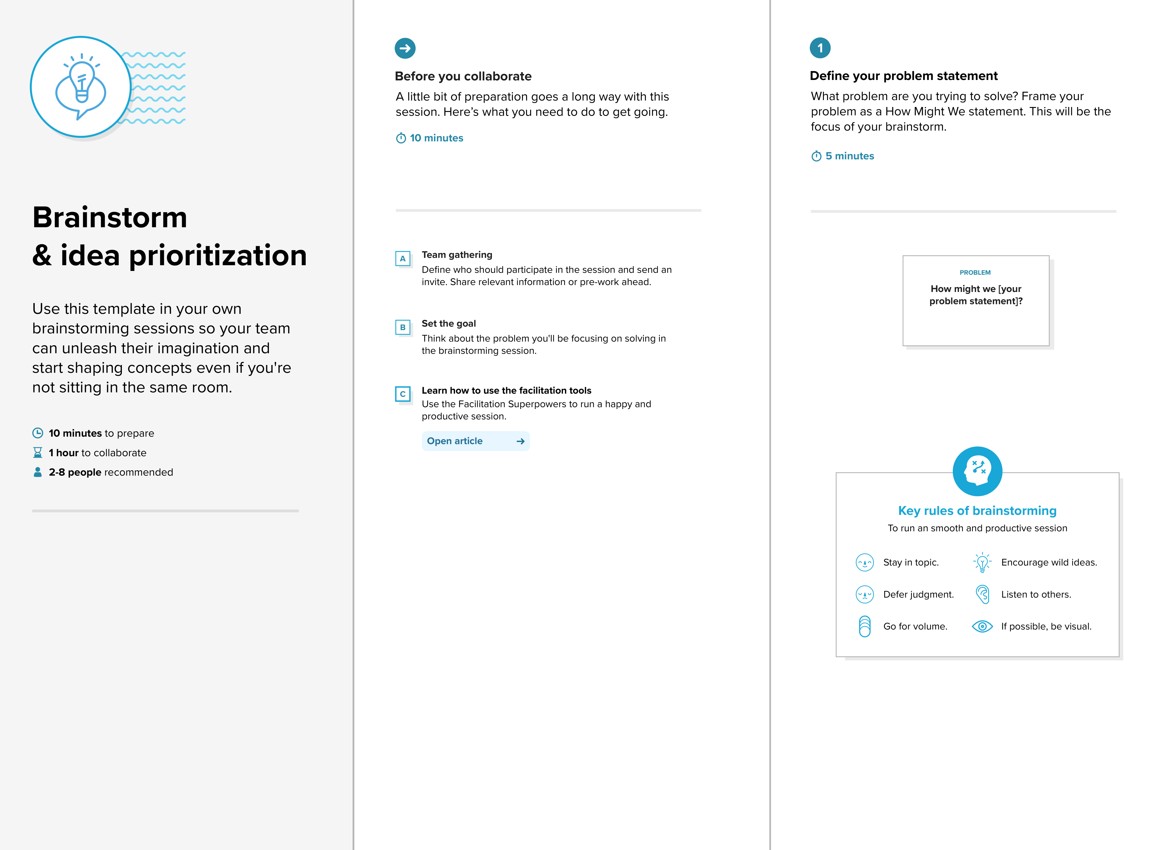
**Brainstorm & Idea Prioritization Template:**

Brainstorming provides a free and open environment that encourages everyone within a team to participate in the creative thinking process that leads to problem solving. Prioritizing volume over value, out-of-the-box ideas are welcome and built upon, and all participants are encouraged to collaborate, helping each other develop a rich amount of creative solutions.

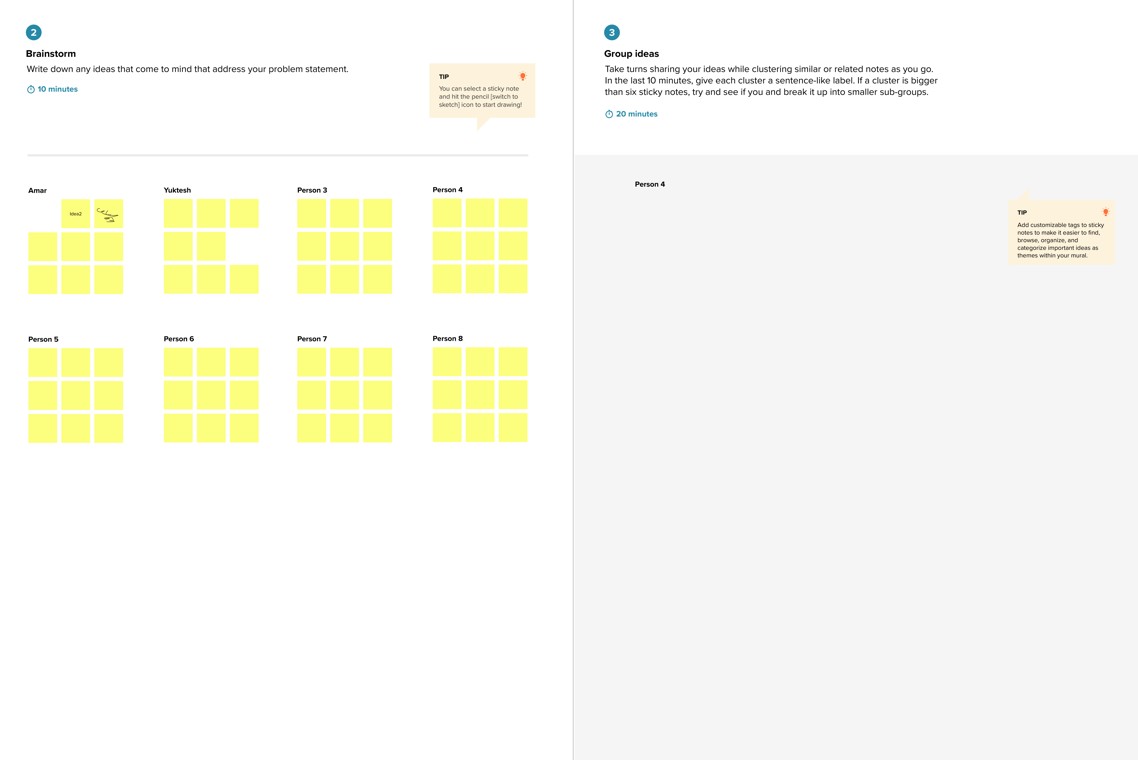
Use this template in your own brainstorming sessions so your team can unleash their imagination and start shaping concepts even if you're not sitting in the same room.

Reference: <https://www.mural.co/templates/brainstorm-and-idea-prioritization>

**Step-1: Team Gathering, Collaboration and Select the Problem Statement**



**Step-2: Brainstorm, Idea Listing and Grouping**



**Step-3: Idea Prioritization**



# Ideation Phase Brainstorm & Idea Prioritization Template

|  |  |
| --- | --- |
| Date | 25 June 2025 |
| Team ID | LTVIP2025TMID33542 |
| Project Name | Grainpalette |
| Maximum Marks | 4 Marks |

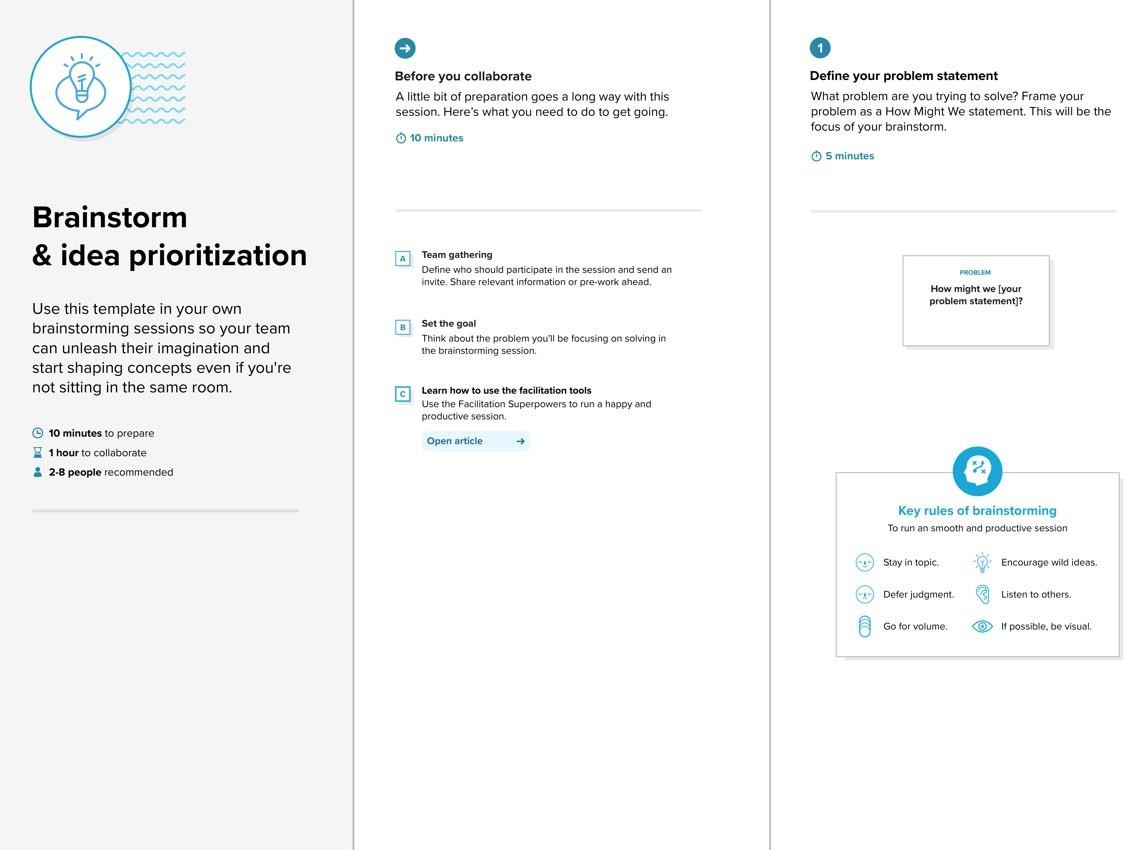
**Brainstorm & Idea Prioritization Template:**

Brainstorming provides a free and open environment that encourages everyone within a team to participate in the creative thinking process that leads to problem solving. Prioritizing volume over value, out-of-the-box ideas are welcome and built upon, and all participants are encouraged to collaborate, helping each other develop a rich amount of creative solutions.

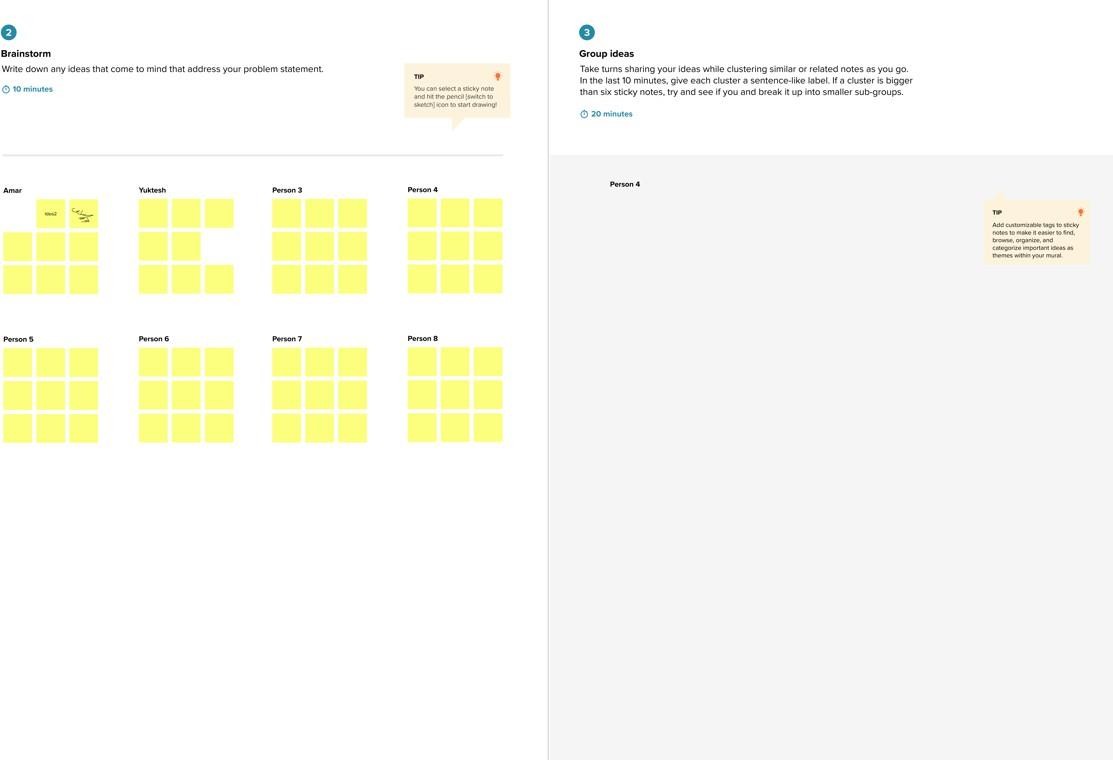
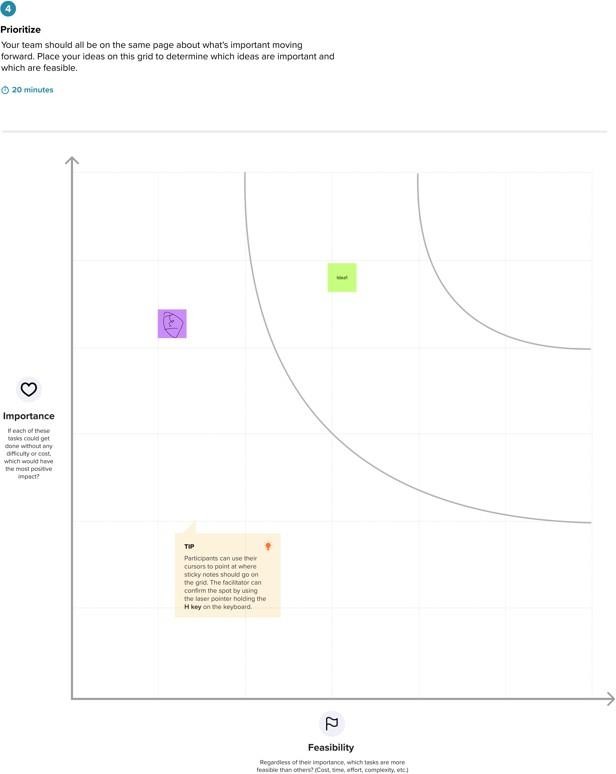
Use this template in your own brainstorming sessions so your team can unleash their imagination and start shaping concepts even if you're not sitting in the same room.

Reference:<https://www.mural.co/templates/brainstorm-and-idea-prioritization>

**Step-1: Team Gathering, Collaboration and Select the Problem Statement**



**Step-2: Brainstorm, Idea Listing and Grouping**



**Step-3: Idea Prioritization**

# Ideation Phase Define the Problem Statements

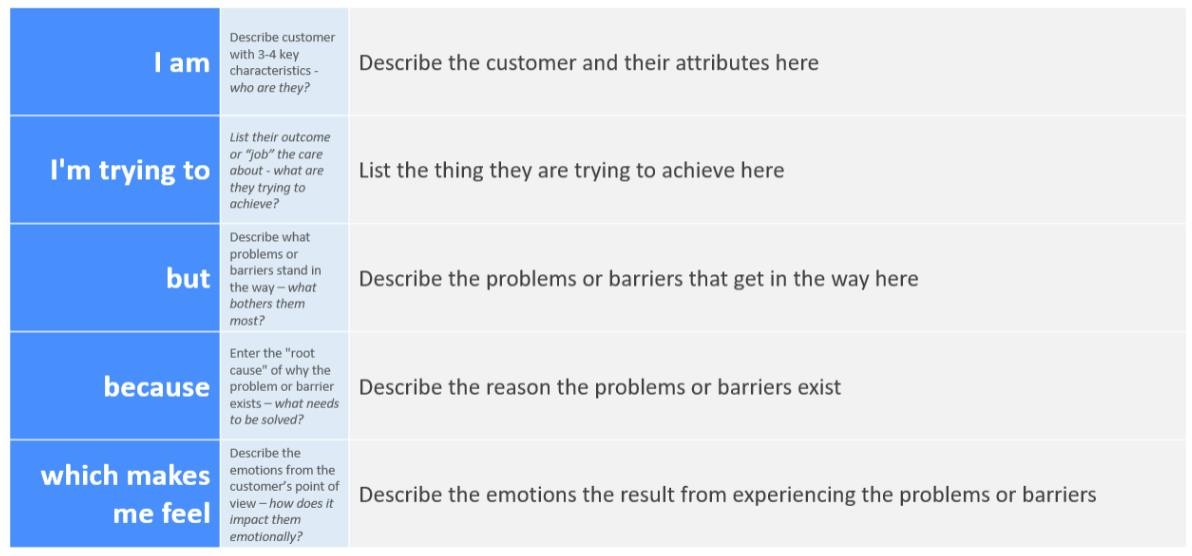
|  |  |
| --- | --- |
| Date | 25 June 2025 |
| Team ID | LTVIP2025TMID33542 |
| Project Name | Grainpalette |
| Maximum Marks | 2 Marks |

**Customer Problem Statement Template:**

Create a problem statement to understand your customer's point of view. The Customer Problem Statement template helps you focus on what matters to create experiences people will love.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Problem**  **Statement (PS)** | **I am**  **(Customer)** | **I’m trying to** | **But** | **Because** | **Which makes me feel** |
| PS-1 |  |  |  |  |  |
| PS-2 |  |  |  |  |  |

A well-articulated customer problem statement allows you and your team to find the ideal solution for the challenges your customers face. Throughout the process, you’ll also be able to empathize with your customers, which helps you better understand how they perceive your product or service.



Reference: <https://miro.com/templates/customer-problem-statement/>**Example:**



# Ideation Phase Define the Problem Statements

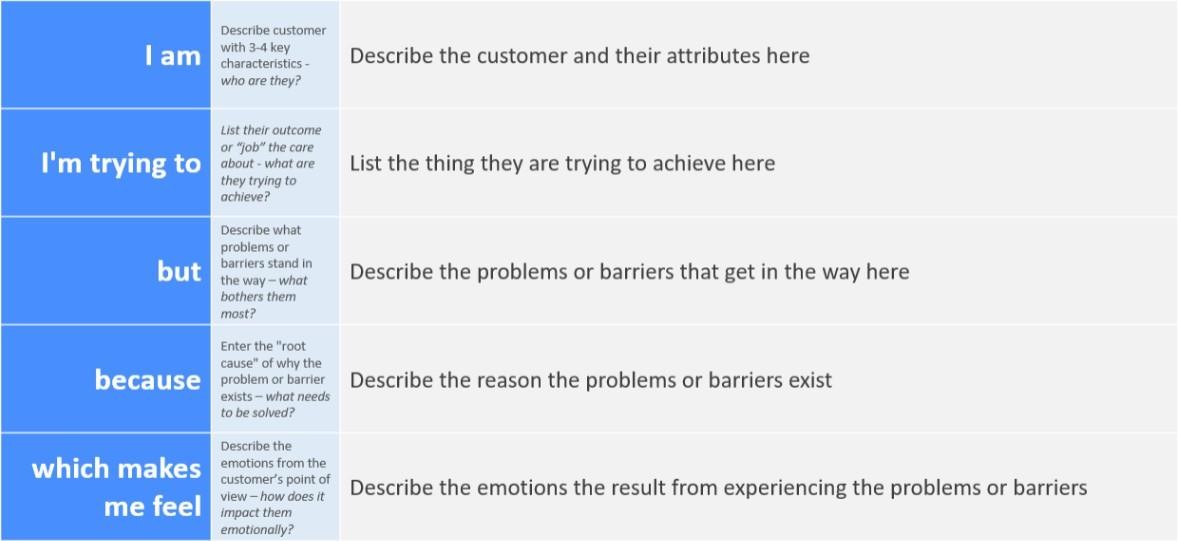
|  |  |
| --- | --- |
| Date | 25 June 2025 |
| Team ID | LTVIP2025TMID33542 |
| Project Name | Grainpalette |
| Maximum Marks | 2 Marks |

**Customer Problem Statement Template:**

Create a problem statement to understand your customer's point of view. The Customer Problem Statement template helps you focus on what matters to create experiences people will love.

A well-articulated customer problem statement allows you and your team to find the ideal solution for the challenges your customers face. Throughout the process, you’ll also be able to empathize with your customers, which helps you better understand how they perceive your product or service.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Problem**  **Statement (PS)** | **I am**  **(Customer)** | **I’m trying to** | **But** | **Because** | **Which makes me feel** |
| PS-1 |  |  |  |  |  |
| PS-2 |  |  |  |  |  |



Reference:<https://miro.com/templates/customer-problem-statement/>

**Example:**



# Ideation Phase Empathize & Discover

|  |  |
| --- | --- |
| Date | 25 June 2025 |
| Team ID | LTVIP2025TMID33542 |
| Project Name | Grainpalette |
| Maximum Marks | 4 Marks |

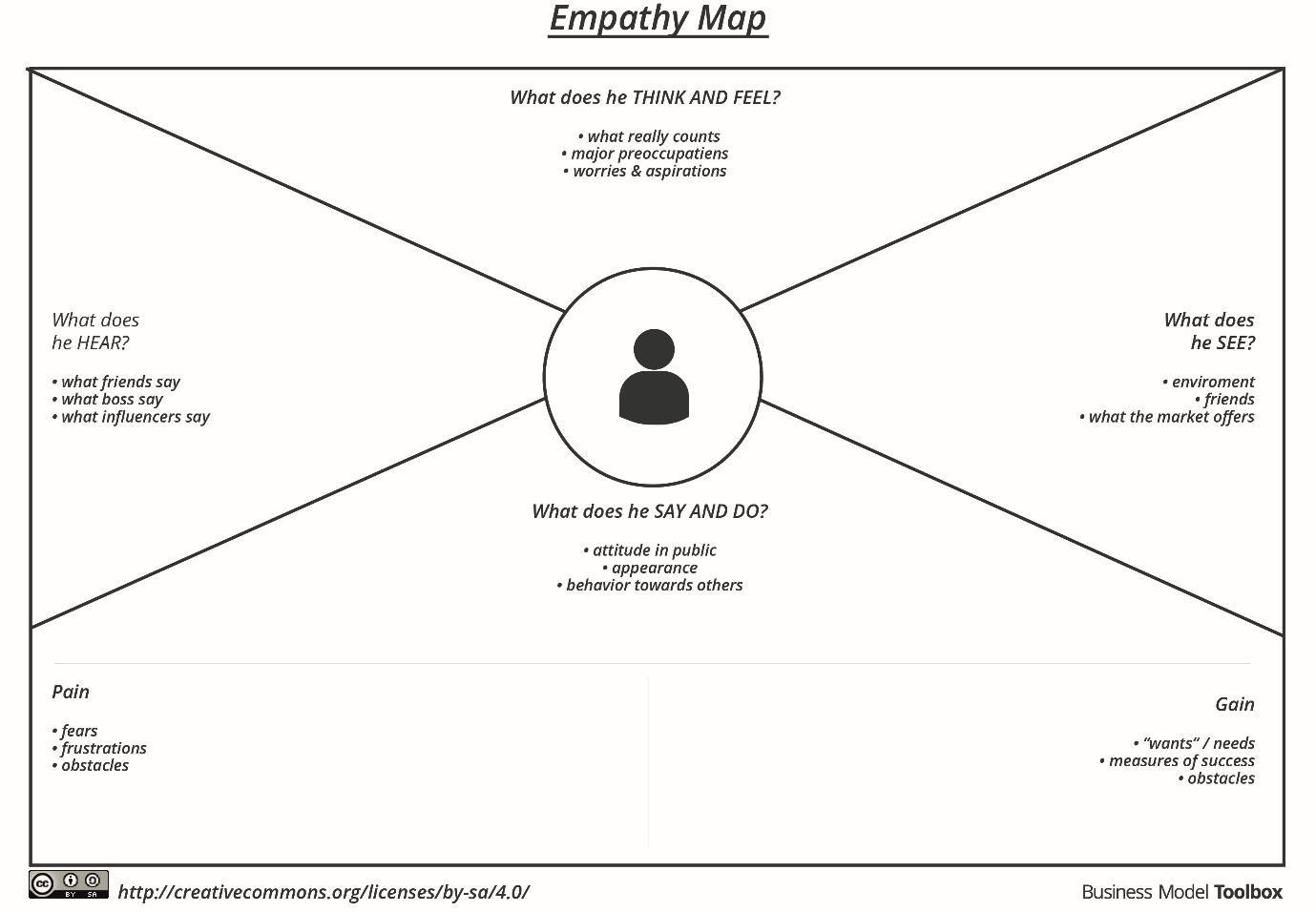
**Empathy Map Canvas:**

An empathy map is a simple, easy-to-digest visual that captures knowledge about a user’s behaviours and attitudes.

It is a useful tool to helps teams better understand their users.

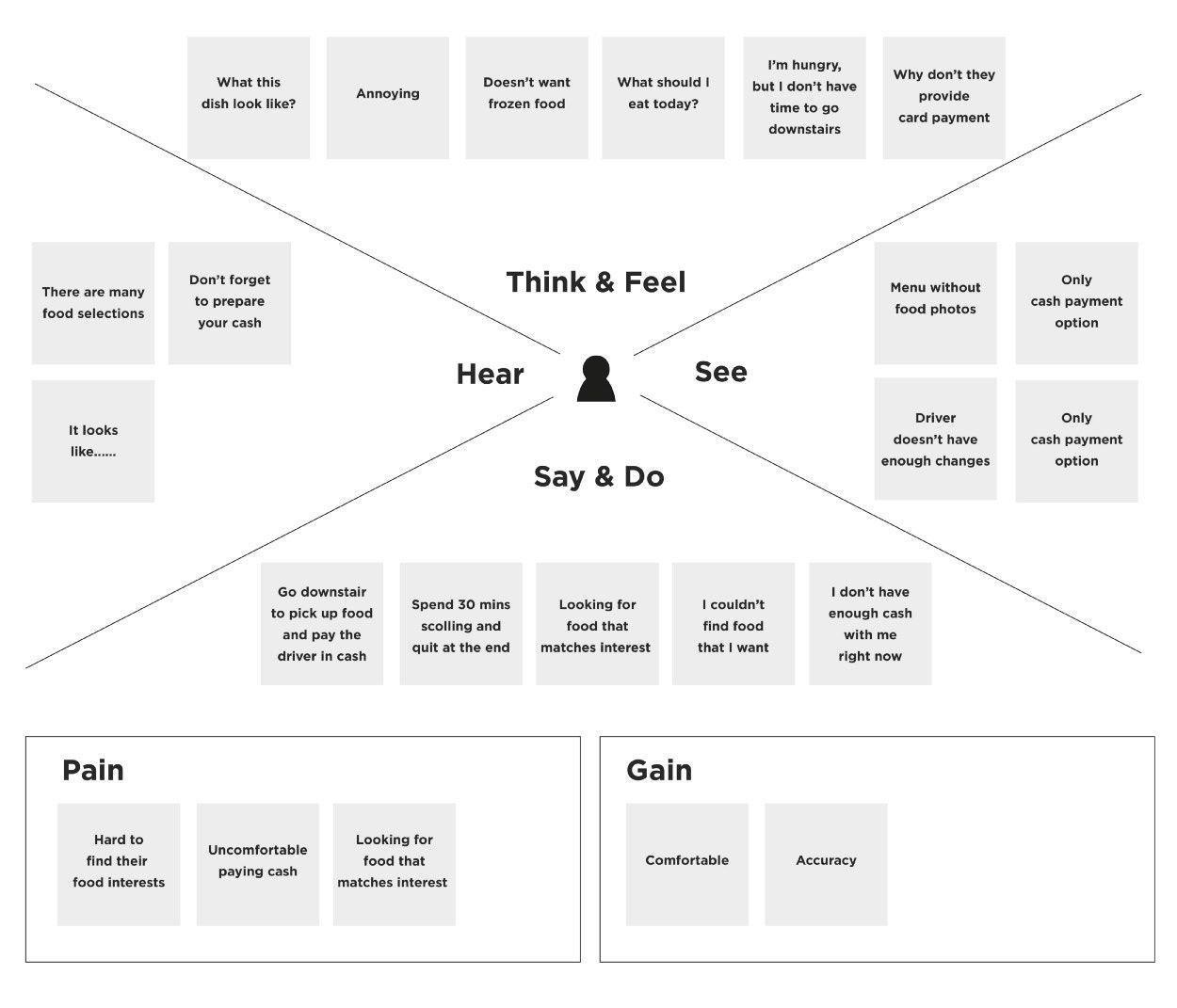
Creating an effective solution requires understanding the true problem and the person who is experiencing it. The exercise of creating the map helps participants consider things from the user’s perspective along with his or her goals and challenges.

**Example:**



Reference: <https://www.mural.co/templates/empathy-map-canvas>

**Example: Food Ordering & Delivery Application**



# Ideation Phase Empathize & Discover

|  |  |
| --- | --- |
| Date | 25 June 2025 |
| Team ID | LTVIP2025TMID33542 |
| Project Name | Grainpalette |
| Maximum Marks | 4 Marks |

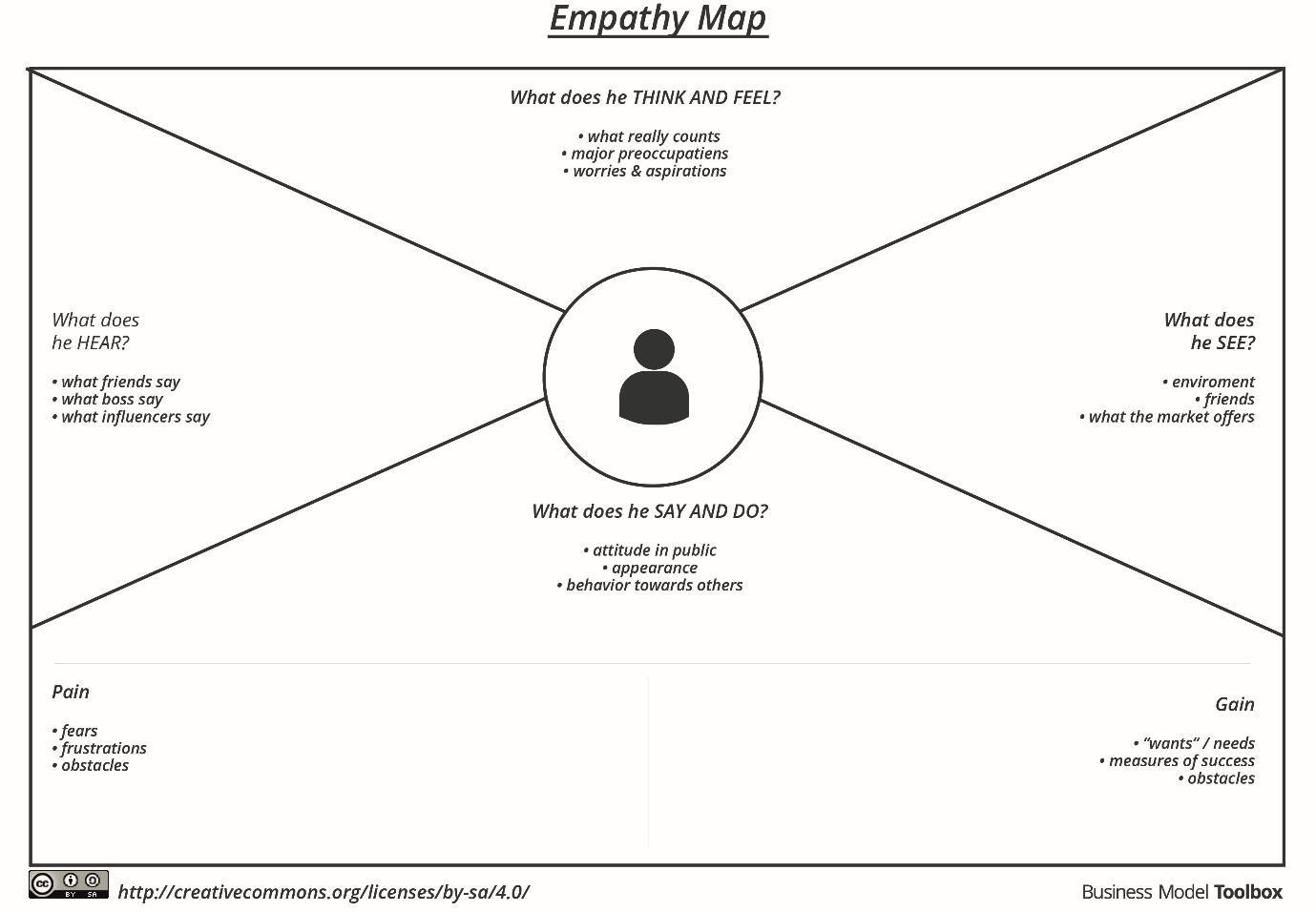
**Empathy Map Canvas:**

An empathy map is a simple, easy-to-digest visual that captures knowledge about a user’s behaviours and attitudes.

It is a useful tool to helps teams better understand their users.

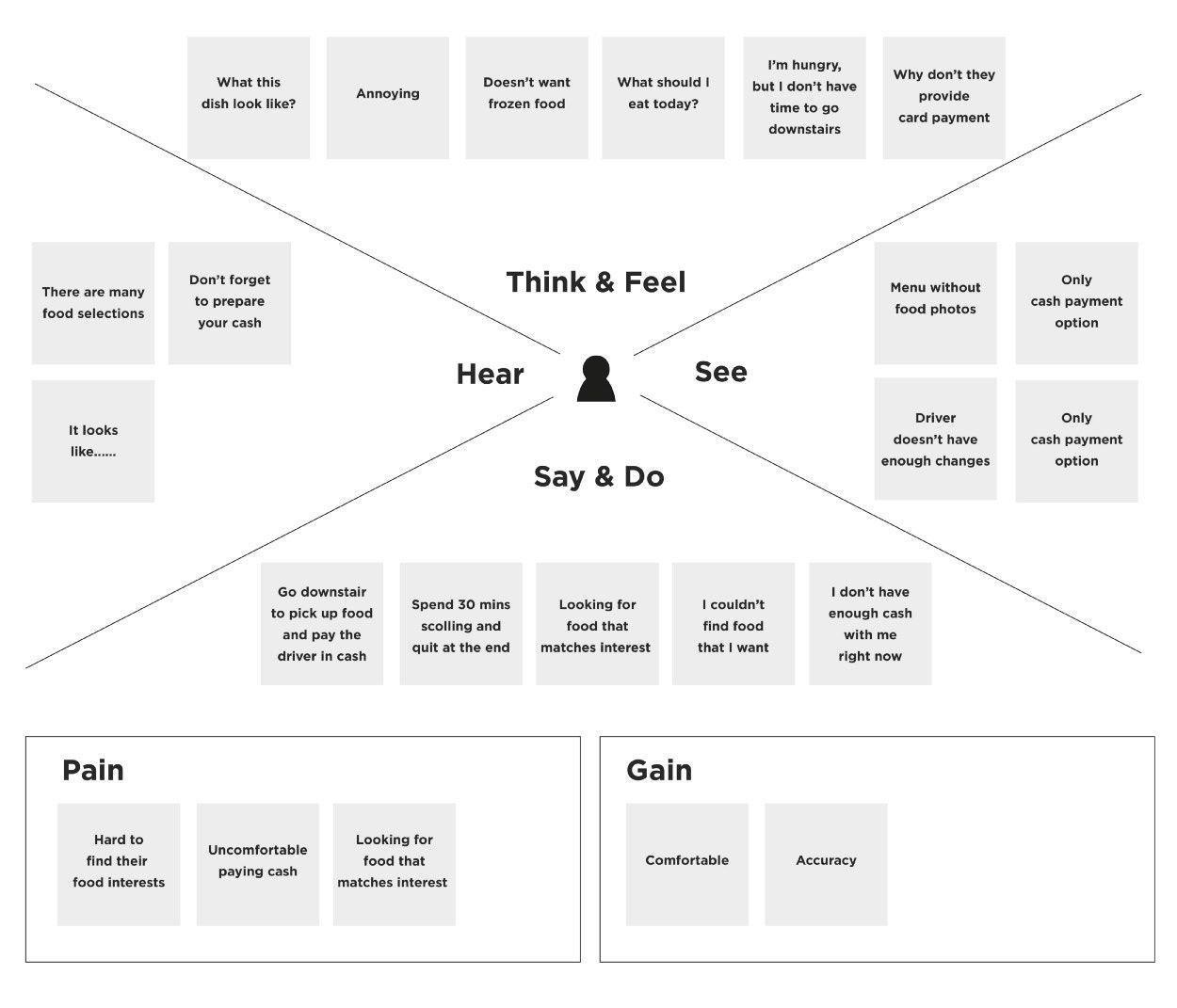
Creating an effective solution requires understanding the true problem and the person who is experiencing it. The exercise of creating the map helps participants consider things from the user’s perspective along with his or her goals and challenges.

**Example:**



Reference: <https://www.mural.co/templates/empathy-map-canvas>

**Example: Food Ordering & Delivery Application**



# Ideation Phase Empathize & Discover

|  |  |
| --- | --- |
| Date | 25 June 2025 |
| Team ID | LTVIP2025TMID33542 |
| Project Name | Grainpalette |
| Maximum Marks | 4 Marks |

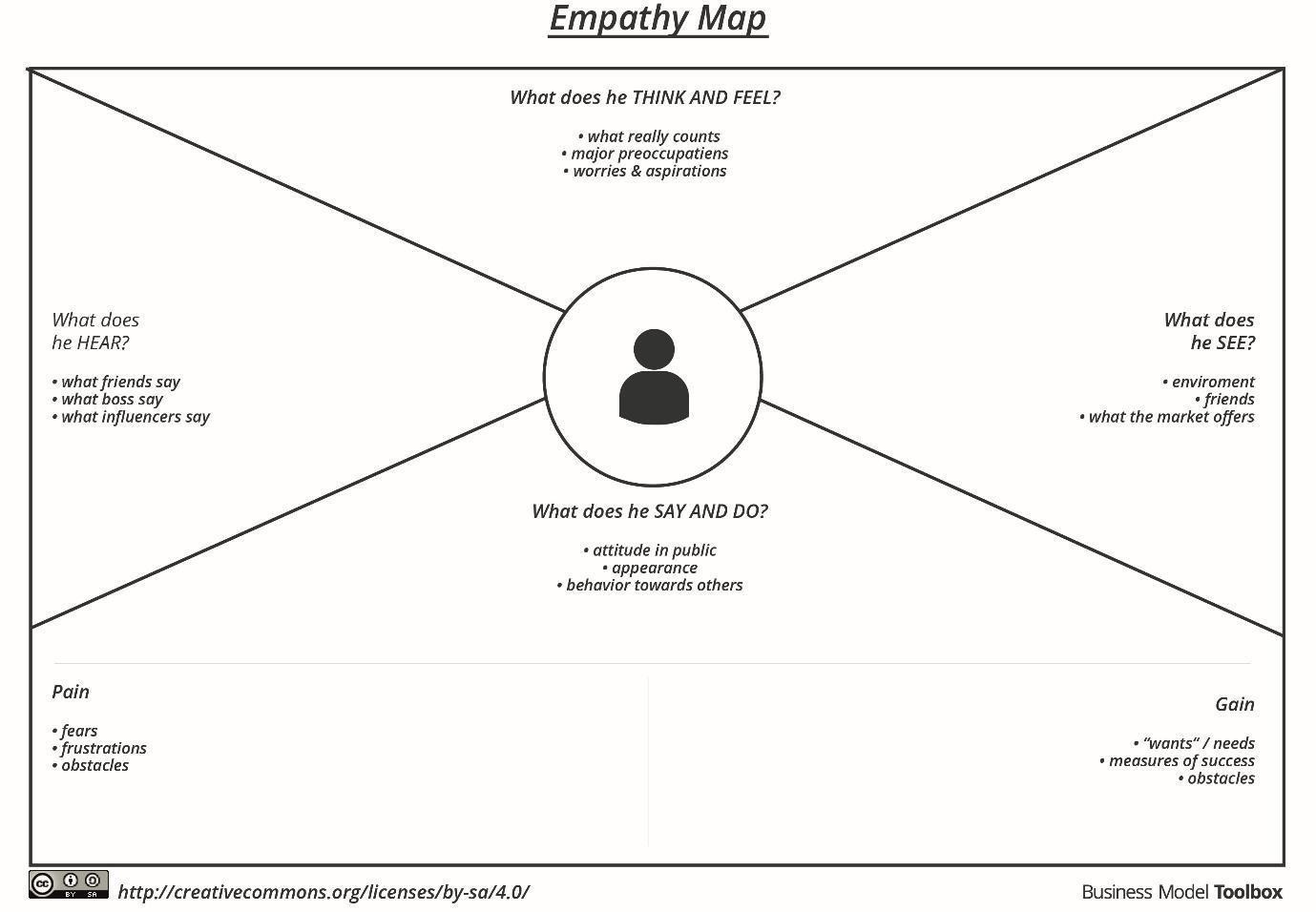
**Empathy Map Canvas:**

An empathy map is a simple, easy-to-digest visual that captures knowledge about a user’s behaviours and attitudes.

It is a useful tool to helps teams better understand their users.

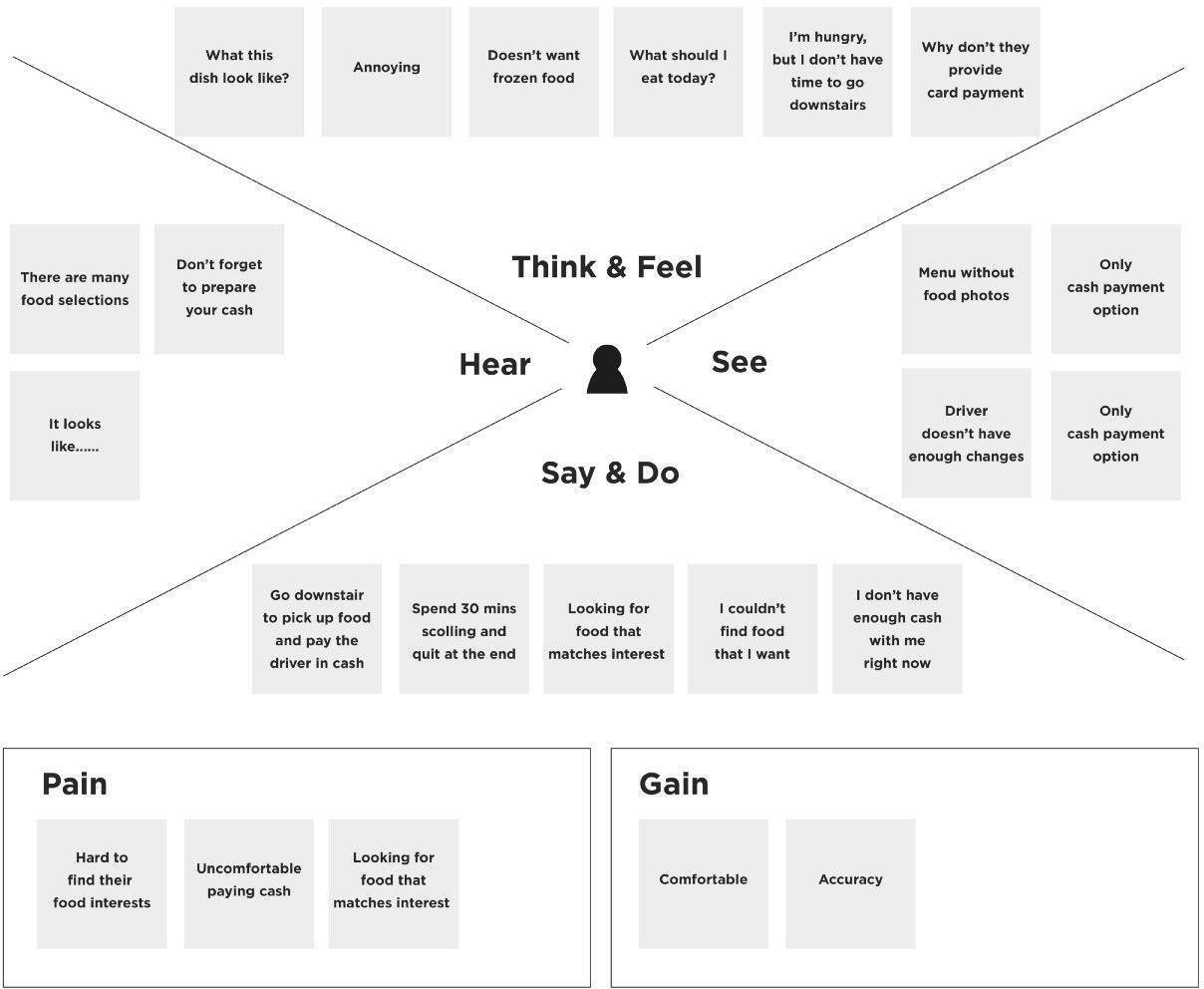
Creating an effective solution requires understanding the true problem and the person who is experiencing it. The exercise of creating the map helps participants consider things from the user’s perspective along with his or her goals and challenges.

**Example:**



Reference:<https://www.mural.co/templates/empathy-map-canvas>

**Example: Food Ordering & Delivery Application**



**Project Development Phase**

**Model Performance Test**

|  |  |
| --- | --- |
| Date | 25 June 2025 |
| Team ID | LTVIP2025TMID33542 |
| Project Name | Grainpalette |
| Maximum Marks |  |

**Model Performance Testing:**

Project team shall fill the following information in model performance testing template.

|  |  |  |  |
| --- | --- | --- | --- |
| **S.No.** | **Parameter** | **Values** | **Screenshot** |
| 1. | Model Summary | **-** |  |
| 2. | Accuracy | Training Accuracy -    Validation Accuracy - |  |
| 3. | Fine Tunning Result( if Done) | Validation Accuracy - |  |

**Functional & Performance Testing Template**

**Model Performance Test**

|  |  |
| --- | --- |
| Date | 25 June 2025 |
| Team ID | LTVIP2025TMID41075 |
| Project Name | Transfer Learning- Based Classification Of Poultry Diseases For Enhanced Health |
| Maximum Marks |  |

**Test Scenarios & Results**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Test**  **Case**  **ID** | **Scenario (What to test)** | **Test Steps (How**  **to test)** | **Expected Result** | **Actual**  **Result** | **Pass/Fail** |
| **FT-01** | Text Input Validation (e.g., topic, job title) | Enter valid and invalid text in input fields | Valid inputs accepted, errors for invalid inputs |  |  |
| **FT-02** | Number Input  Validation (e.g., word count, size, rooms) | Enter numbers within and outside the valid range | Accepts valid values, shows error for outof-range |  |  |
| **FT-03** | Content Generation (e.g., blog, resume, design idea) | Provide complete inputs and click "Generate" | Correct content is generated based on  input |  |  |
| **FT-04** | API Connection Check | Check if API key is correct and model responds | API responds successfully |  |  |
| **PT-01** | Response Time Test | Use a timer to check content generation time | Should be under 3 seconds |  |  |
| **PT-02** | API Speed Test | Send multiple API calls at the same time | API should not slow down |  |  |
| **PT-03** | File Upload Load Test (e.g., PDFs) | Upload multiple PDFs and check processing | Should work smoothly without crashing |  |  |

**Project Development Phase**

**Model Performance Test**

|  |  |
| --- | --- |
| Date | 25 June 2025 |
| Team ID | LTVIP2025TMID33542 |
| Project Name | Grainpalette |
| Maximum Marks | 10 Marks |

**Model Performance Testing:**

Project team shall fill the following information in model performance testing template.

|  |  |  |  |
| --- | --- | --- | --- |
| **S.No.** | **Parameter** | **Values** | **Screenshot** |
| 1. | Metrics | **Regression Model:**  MAE - , MSE - , RMSE - , R2 score -    **Classification Model:**  Confusion Matrix - , Accuray Score- & Classification Report - |  |
| 2. | Tune the Model | Hyperparameter Tuning - Validation Method - |  |

**Project Development Phase**

**Model Performance Test**

|  |  |
| --- | --- |
| Date | 25 June 2025 |
| Team ID | LTVIP2025TMID33542 |
| Project Name | Grainpalette |
| Maximum Marks |  |

**Model Performance Testing:**

Project team shall fill the following information in model performance testing template.

|  |  |  |
| --- | --- | --- |
| **S.No.** | **Parameter** | **Screenshot / Values** |
| 1. | Data Rendered |  |
| 2. | Data Preprocessing |  |
| 3. | Utilization of Data Filters |  |
| 4. | DAX Queries Used |  |
| 5. | Dashboard design | No of Visualizations / Graphs - |
| 6 | Report Design | No of Visualizations / Graphs - |

**Project Development Phase**

**Model Performance Test**

|  |  |
| --- | --- |
| Date | 25 June 2025 |
| Team ID | LTVIP2025TMID33542 |
| Project Name | Grainpalette |
| Maximum Marks |  |

**Model Performance Testing:**

Project team shall fill the following information in model performance testing template.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **S.No.** | **Parameter** | **Values** |  | **Screenshot** |
| 1. | Model Summary | Salesforce automation setup for Data management using Object, Fields and Reports.    **Note :** Import Records if data  Match Correctly then Records will  Created or Else it will Show Error |  |  |
| 2. | Accuracy | Training Accuracy - 98%    Validation Accuracy - 98% | . |  |
| 3. | Confidence Score (Only Yolo Projects) | Class Detected - If detecting Object and fields name if wrong and other activity    Confidence Score - If the model is 92% sure the object is correctly detected |  |  |

# User Acceptance Testing (UAT) Template

|  |  |
| --- | --- |
| Date | 25 June 2025 |
| Team ID | LTVIP2025TMID33542 |
| Project Name | Grainpalette |
| Maximum Marks |  |

**Project Overview:**

Project Name: [Enter Project Name]

Project Description: [Brief Description of the Project]

Project Version: [Version Number] Testing Period: [Start Date] to [End Date]

**Testing Scope:**

[List of Features and Functionalities to be Tested] [List of User Stories or Requirements to be Tested] **Testing Environment:**

URL/Location: [Web URL or Application Location]

Credentials (if required): [Username/Password] **Test Cases:**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Test Case**  **ID** | **Test**  **Scenario** | **Test Steps** | **Expected Result** | **Actual Result** | **Pass/Fail** |
| TC-001 | [Describe the scenario to be tested] | [Step 1]  [Step 2]  [Step 3] | [Describe the expected outcome] | [Record the actual outcome] | [Pass/Fail] |
| ... | ... | ... | ... | ... | ... |

**Bug Tracking:**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Bug ID** | **Bug**  **Description** | **Steps to reproduce** | **Severity** | **Status** | **Additional feedback** |
| BG-001 | [Describe the issue or | [Step 1]  [Step 2] | [Low/Medi | [Open/In  Progress/ | [Any additional |
|  | bug encountered  ] | [Step 3] | um/High] | Closed] | comments or feedback] |
| ... | ... | ... | ... | ... | ... |

**Sign-off:**

Tester Name: [Name of Tester]

Date: [Date of Test Completion]

Signature: [Tester's Signature]

**Notes:**

* Ensure that all test cases cover both positive and negative scenarios.
* Encourage testers to provide detailed feedback, including any suggestions for improvement.
* Bug tracking should include details such as severity, status, and steps to reproduce.
* Obtain sign-off from both the project manager and product owner before proceeding with deployment.

**Project Design Phase**

**Problem – Solution Fit Template**

|  |  |
| --- | --- |
| Date | 25 June 2025 |
| Team ID | LTVIP2025TMID33542 |
| Project Name | Grainpalette |
| Maximum Marks | 2 Marks |

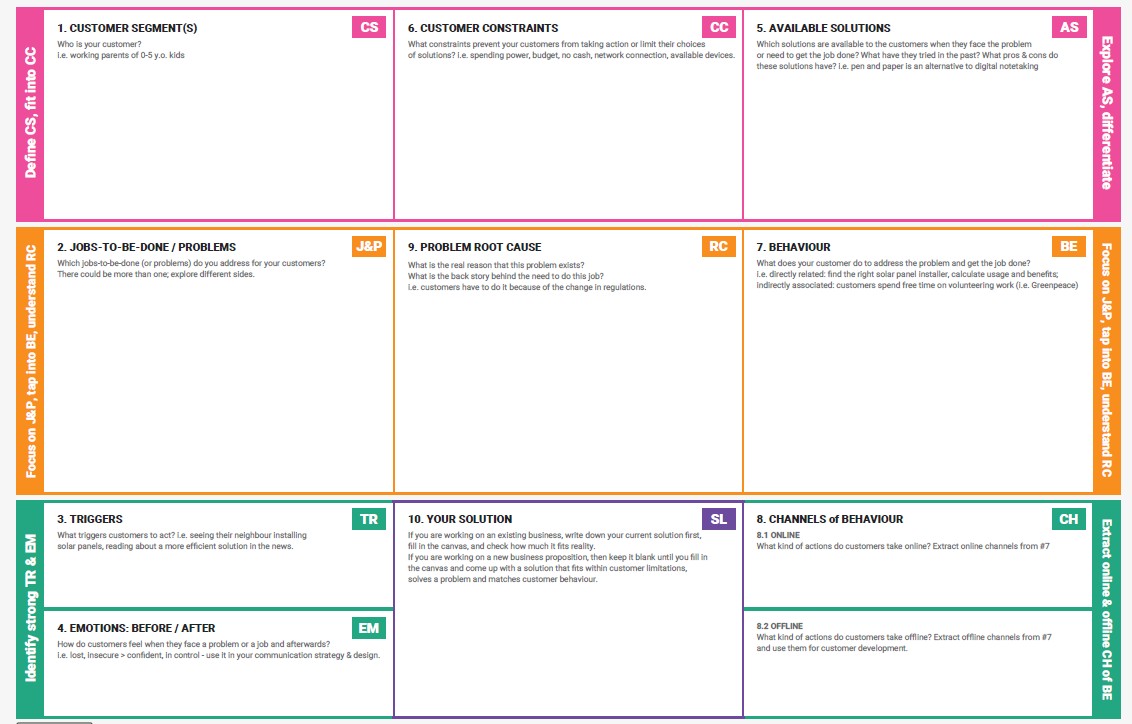
**Problem – Solution Fit Template:**

The Problem-Solution Fit simply means that you have found a problem with your customer and that the solution you have realized for it actually solves the customer’s problem. It helps entrepreneurs, marketers and corporate innovators identify behavioral patterns and recognize what would work and why

**Purpose:**

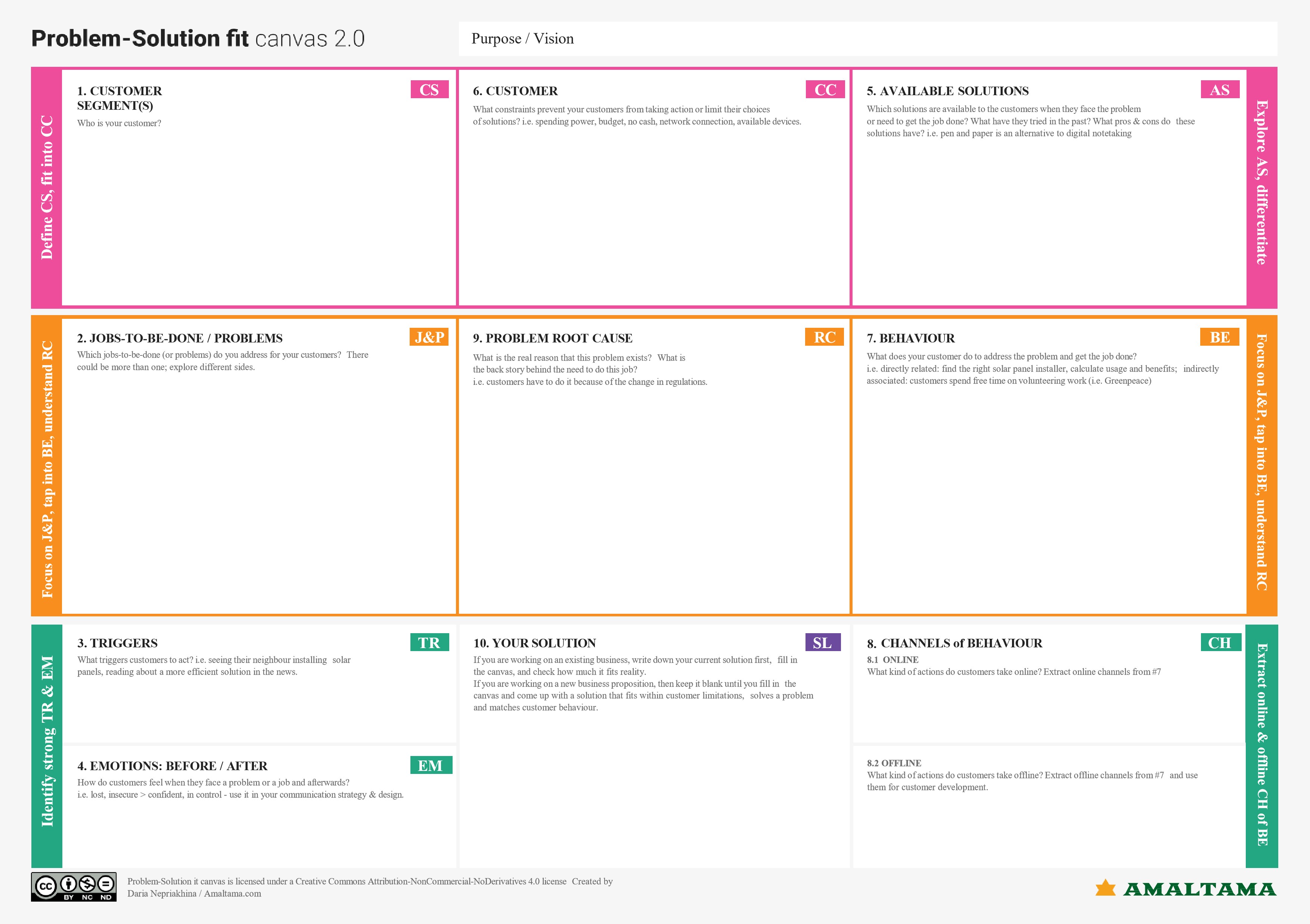
|  |  |
| --- | --- |
| ❑ | Solve complex problems in a way that fits the state of your customers. |
| ❑ | Succeed faster and increase your solution adoption by tapping into existing mediums and channels of behavior. |
| ❑ | Sharpen your communication and marketing strategy with the right triggers and messaging. |
| ❑ | Increase touch-points with your company by finding the right problem-behavior fit and building trust by solving frequent annoyances, or urgent or costly problems. |
| ❑ | **Understand the existing situation in order to improve it for your target group.** |

**Template:**



References:

1. <https://www.ideahackers.network/problem-solution-fit-canvas/>
2. <https://medium.com/@epicantus/problem-solution-fit-canvas-aa3dd59cb4fe>



**A Sprint** fixed period or duration in which a team works to complete a set of tasks

An **Epic** is a **big task or project** that is too large to complete in one sprint. It is broken down into **smaller tasks (stories)** that can be completed over multiple sprints.

A **Story** is a small task . It is part of an **Epic**.

A **Story Point** is a number that represents how much effort a story takes to complete.

(usually in form of Fibonacci series)

1. Very Easy task
2. Easy task
3. Moderate task **5-** Difficult task

**Sprint 1: (5 Days)**

Data Collection

Collection of Data **2**

Loading Data **1**

Data Preprocessing

Handling Missing Values **3**

Handling Categorical values **2**

**Sprint 2 (5 Days)**

Model Building

Model Building **5**

Testing Model **3**

Deployment

Working HTML Pages **3**

Flask deployment **5**

**Total Story Points**

Sprint 1 = 8

Sprint 2 = 16

Velocity= Total Story Points Completed/ Number of Sprints

Total story Points= 16+8 =24

No of Sprints= 2

**Velocity** = (16+8)/2= 24/2

12 (Story Points per Sprint)

**Your team’s velocity is 12 Story Points per Sprint.**

**Project Planning Phase**

**Project Planning Template (Product Backlog, Sprint Planning, Stories, Story points)**

|  |  |
| --- | --- |
| Date | 25 June 2025 |
| Team ID | LTVIP2025TMID33542 |
| Project Name | Grainpalette |
| Maximum Marks | 5 Marks |

**Product Backlog, Sprint Schedule, and Estimation (4 Marks)**

Use the below template to create product backlog and sprint schedule

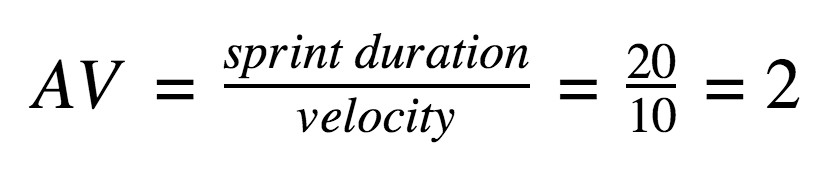
|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Sprint** | **Functional**  **Requirement (Epic)** | **User Story Number** | **User Story / Task** | **Story Points** | **Priority** | **Team**  **Members** |
| Sprint-1 | Registration | USN-1 | As a user, I can register for the application by entering my email, password, and confirming my password. | 2 | High |  |
| Sprint-1 |  | USN-2 | As a user, I will receive confirmation email once I have registered for the application | 1 | High |  |
| Sprint-2 |  | USN-3 | As a user, I can register for the application through Facebook | 2 | Low |  |
| Sprint-1 |  | USN-4 | As a user, I can register for the application through Gmail | 2 | Medium |  |
| Sprint-1 | Login | USN-5 | As a user, I can log into the application by entering email & password | 1 | High |  |
|  | Dashboard |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |

**Project Tracker, Velocity & Burndown Chart: (4 Marks)**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Sprint** | **Total Story Points** | **Duration** | **Sprint Start Date** | **Sprint End Date (Planned)** | **Story Points**  **Completed (as on**  **Planned End Date)** | **Sprint Release Date (Actual)** |
| Sprint-1 | 20 | 6 Days | 24 Oct 2022 | 29 Oct 2022 | 20 | 29 Oct 2022 |
| Sprint-2 | 20 | 6 Days | 31 Oct 2022 | 05 Nov 2022 |  |  |
| Sprint-3 | 20 | 6 Days | 07 Nov 2022 | 12 Nov 2022 |  |  |
| Sprint-4 | 20 | 6 Days | 14 Nov 2022 | 19 Nov 2022 |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |

**Velocity:**

Imagine we have a 10-day sprint duration, and the velocity of the team is 20 (points per sprint). Let’s calculate the team’s average velocity (AV) per iteration unit (story points per day)



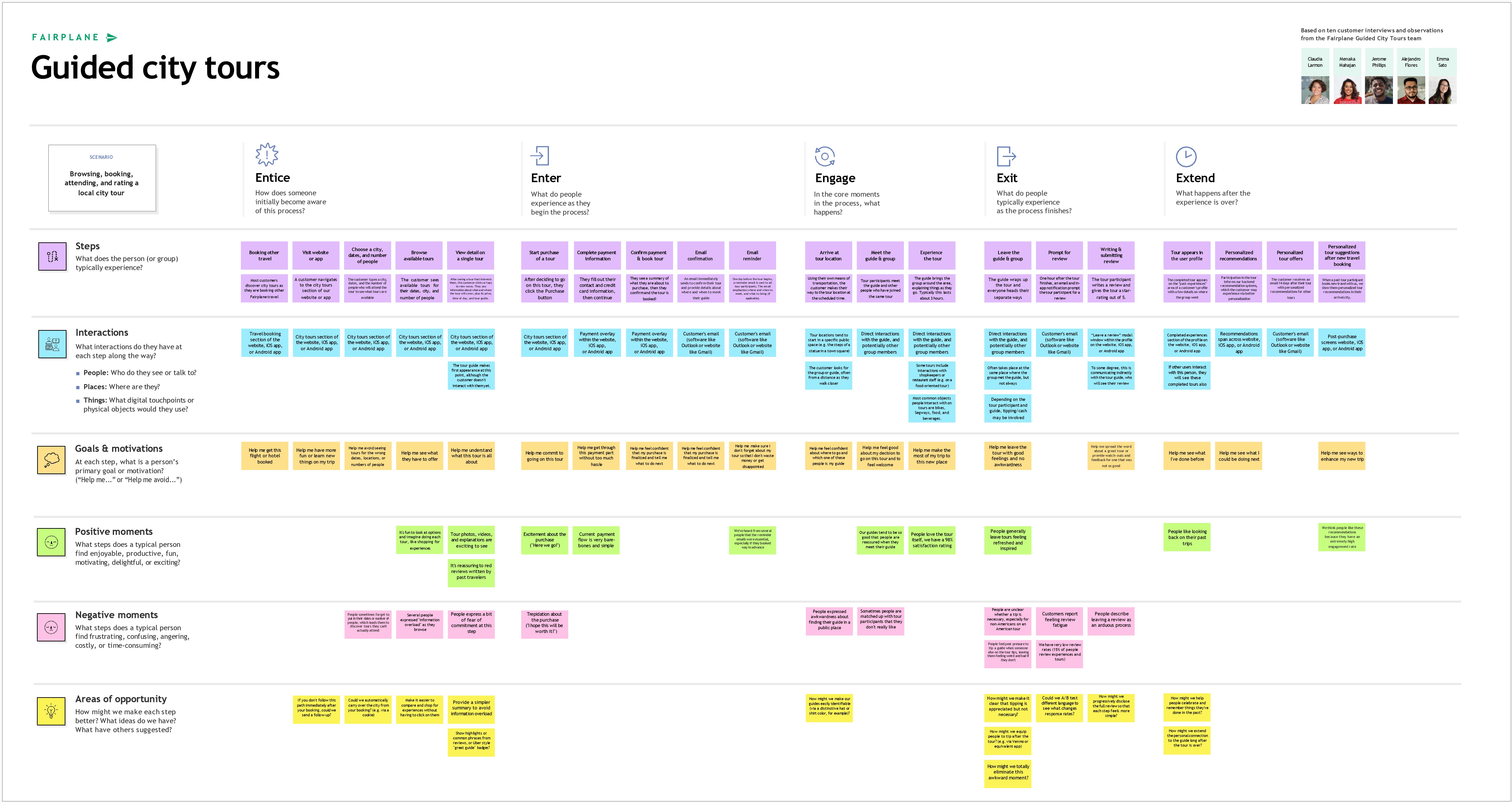
**Burndown Chart:**

A burn down chart is a graphical representation of work left to do versus time. It is often used in agile [software development](https://www.visual-paradigm.com/scrum/what-is-agile-software-development/) methodologies such as [Scrum.](https://www.visual-paradigm.com/scrum/scrum-in-3-minutes/) However, burn down charts can be applied to any project containing measurable progress over time.

[**https://www.visual-paradigm.com/scrum/scrum-burndown-chart/**](https://www.visual-paradigm.com/scrum/scrum-burndown-chart/) [**https://www.atlassian.com/agile/tutorials/burndown-charts**](https://www.atlassian.com/agile/tutorials/burndown-charts)

**Reference:** [**https://www.atlassian.com/agile/project-management**](https://www.atlassian.com/agile/project-management) [**https://www.atlassian.com/agile/tutorials/how-to-do-scrum-with-jira-software**](https://www.atlassian.com/agile/tutorials/how-to-do-scrum-with-jira-software) [**https://www.atlassian.com/agile/tutorials/epics**](https://www.atlassian.com/agile/tutorials/epics) [**https://www.atlassian.com/agile/tutorials/sprints**](https://www.atlassian.com/agile/tutorials/sprints) [**https://www.atlassian.com/agile/project-management/estimation**](https://www.atlassian.com/agile/project-management/estimation)

[**https://www.atlassian.com/agile/tutorials/burndown-charts**](https://www.atlassian.com/agile/tutorials/burndown-charts)



**Project Design Phase-II**

**Data Flow Diagram & User Stories**

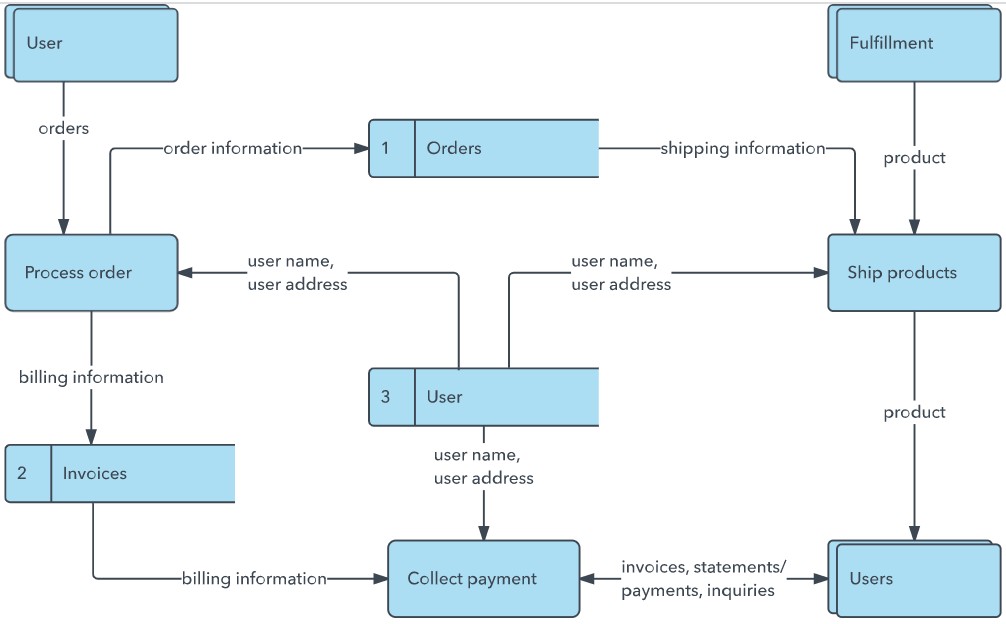
|  |  |
| --- | --- |
| Date | 25 June 2025 |
| Team ID | LTVIP2025TMID33542 |
| Project Name | Grainpalette |
| Maximum Marks | 4 Marks |

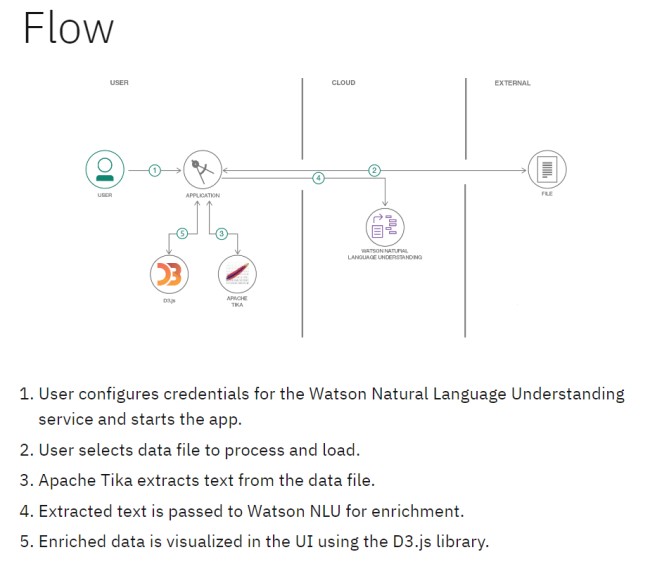
**Data Flow Diagrams:**

A Data Flow Diagram (DFD) is a traditional visual representation of the information flows within a system. A neat and clear DFD can depict the right amount of the system requirement graphically. It shows how data enters and leaves the system, what changes the information, and where data is stored.

Example: DFD Level 0 (Industry Standard)

**Example:** [**(Simplified)**](https://developer.ibm.com/patterns/visualize-unstructured-text/)





**User Stories**

Use the below template to list all the user stories for the product.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **User Type** | **Functional**  **Requirement**  **(Epic)** | **User Story Number** | **User Story / Task** | **Acceptance criteria** | **Priority** | **Release** |
| Customer  (Mobile user) | Registration | USN-1 | As a user, I can register for the application by entering my email, password, and confirming my password. | I can access my account / dashboard | High | Sprint-1 |
|  |  | USN-2 | As a user, I will receive confirmation email once I have registered for the application | I can receive confirmation email & click confirm | High | Sprint-1 |
|  |  | USN-3 | As a user, I can register for the application through Facebook | I can register & access the dashboard with Facebook  Login | Low | Sprint-2 |
|  |  | USN-4 | As a user, I can register for the application through Gmail |  | Medium | Sprint-1 |
|  | Login | USN-5 | As a user, I can log into the application by entering email & password |  | High | Sprint-1 |
|  | Dashboard |  |  |  |  |  |
| Customer (Web user) |  |  |  |  |  |  |
| Customer Care Executive |  |  |  |  |  |  |
| Administrator |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |

**Project Design Phase-II**

**Solution Requirements (Functional & Non-functional)**

|  |  |
| --- | --- |
| Date | 25 June 2025 |
| Team ID | LTVIP2025TMID33542 |
| Project Name | Grainpalette |
| Maximum Marks | 4 Marks |

**Functional Requirements:**

Following are the functional requirements of the proposed solution.

|  |  |  |
| --- | --- | --- |
| **FR No.** | **Functional Requirement (Epic)** | **Sub Requirement (Story / Sub-Task)** |
| FR-1 | User Registration | Registration through Form  Registration through Gmail  Registration through LinkedIN |
| FR-2 | User Confirmation | Confirmation via Email Confirmation via OTP |
| FR-3 |  |  |
| FR-4 |  |  |
|  |  |  |
|  |  |  |

**Non-functional Requirements:**

Following are the non-functional requirements of the proposed solution.

|  |  |  |
| --- | --- | --- |
| **FR No.** | **Non-Functional Requirement** | **Description** |
| NFR-1 | **Usability** |  |
| NFR-2 | **Security** |  |
| NFR-3 | **Reliability** |  |
| NFR-4 | **Performance** |  |
| NFR-5 | **Availability** |  |
| NFR-6 | **Scalability** |  |

**Project Design Phase-II**

**Solution Requirements (Functional & Non-functional)**

|  |  |
| --- | --- |
| Date | 25 June 2025 |
| Team ID | LTVIP2025TMID33542 |
| Project Name | Grainpalette |
| Maximum Marks | 4 Marks |

**Functional Requirements:**

Following are the functional requirements of the proposed solution.

|  |  |  |
| --- | --- | --- |
| **FR No.** | **Functional Requirement (Epic)** | **Sub Requirement (Story / Sub-Task)** |
| FR-1 | User Registration | Registration through Form  Registration through Gmail  Registration through LinkedIN |
| FR-2 | User Confirmation | Confirmation via Email Confirmation via OTP |
| FR-3 |  |  |
| FR-4 |  |  |
|  |  |  |
|  |  |  |

**Non-functional Requirements:**

Following are the non-functional requirements of the proposed solution.

|  |  |  |
| --- | --- | --- |
| **FR No.** | **Non-Functional Requirement** | **Description** |
| NFR-1 | **Usability** |  |
| NFR-2 | **Security** |  |
| NFR-3 | **Reliability** |  |
| NFR-4 | **Performance** |  |
| NFR-5 | **Availability** |  |
| NFR-6 | **Scalability** |  |

**Project Design Phase-II**

**Technology Stack (Architecture & Stack)**

|  |  |
| --- | --- |
| Date | 25 June 2025 |
| Team ID | LTVIP2025TMID33542 |
| Project Name | Grainpalette |
| Maximum Marks | 4 Marks |

**Technical Architecture:**

The Deliverable shall include the architectural diagram as below and the information as per the table1 & table 2

**Example: Order processing during pandemics for offline mode**

**Reference:** [**https://developer.ibm.com/patterns/ai-powered-backend-system-for-order-processing-during-pandemics/**](https://developer.ibm.com/patterns/ai-powered-backend-system-for-order-processing-during-pandemics/)

**Table**

**-**

**:**

**: Components & Technologies**

**1**

**S.No**

**Component**

**Description**

**Technology**

Guidelines:

Include all the processe

s (As an application logic /

Technology Block)

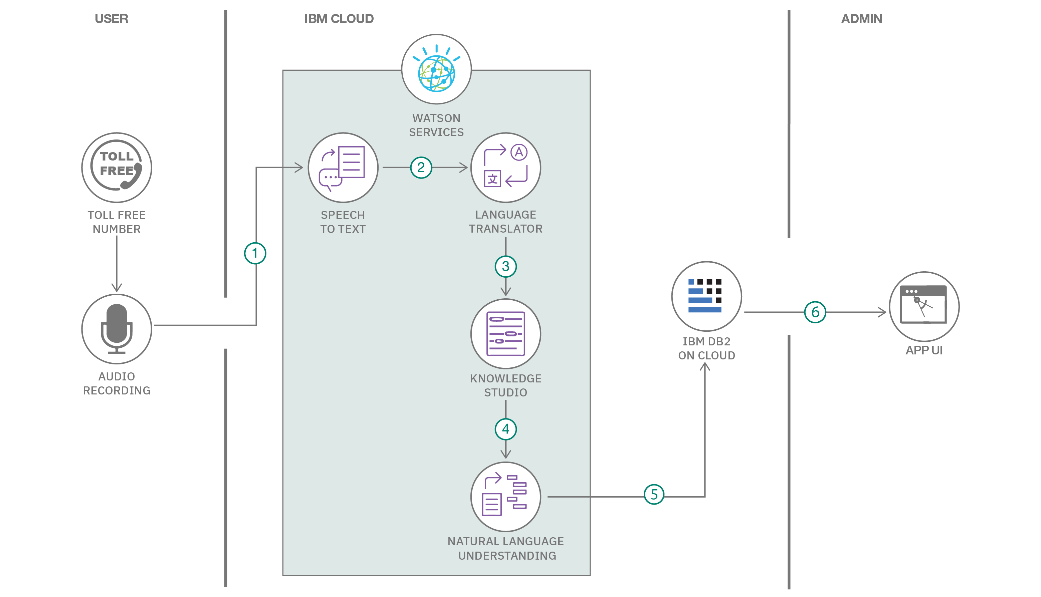
Provide infrastructural demarcation (Local / Cloud)

Indicate external interfaces (third party API’s etc.)

Indicate Data Storage components / services

Indicate interface to machine learning models (if

applicable)



|  |  |  |  |
| --- | --- | --- | --- |
| 1. | User Interface | How user interacts with application e.g. Web UI, Mobile App, Chatbot etc. | HTML, CSS, JavaScript / Angular Js / React Js etc. |
| 2. | Application Logic-1 | Logic for a process in the application | Java / Python |
| 3. | Application Logic-2 | Logic for a process in the application | IBM Watson STT service |
| 4. | Application Logic-3 | Logic for a process in the application | IBM Watson Assistant |
| 5. | Database | Data Type, Configurations etc. | MySQL, NoSQL, etc. |
| 6. | Cloud Database | Database Service on Cloud | IBM DB2, IBM Cloudant etc. |
| 7. | File Storage | File storage requirements | IBM Block Storage or Other Storage Service or Local Filesystem |
| 8. | External API-1 | Purpose of External API used in the application | IBM Weather API, etc. |
| 9. | External API-2 | Purpose of External API used in the application | Aadhar API, etc. |
| 10. | Machine Learning Model | Purpose of Machine Learning Model | Object Recognition Model, etc. |
| 11. | Infrastructure (Server / Cloud) | Application Deployment on Local System / Cloud Local Server Configuration:  Cloud Server Configuration : | Local, Cloud Foundry, Kubernetes, etc. |

|  |  |  |  |
| --- | --- | --- | --- |
| **S.No** | **Characteristics** | **Description** | **Technology** |
| 1. | Open-Source Frameworks | List the open-source frameworks used | Technology of Opensource framework |
| 2. | Security Implementations | List all the security / access controls implemented, use of firewalls etc. | e.g. SHA-256, Encryptions, IAM Controls, OWASP etc. |
| 3. | Scalable Architecture | Justify the scalability of architecture (3 – tier, Micro-services) | Technology used |
| 4. | Availability | Justify the availability of application (e.g. use of load balancers, distributed servers etc.) | Technology used |

**Table-2: Application Characteristics:**

|  |  |  |  |
| --- | --- | --- | --- |
| **S.No** | **Characteristics** | **Description** | **Technology** |
| 5. | Performance | Design consideration for the performance of the application (number of requests per sec, use of Cache, use of CDN’s) etc. | Technology used |

**References:**

[**https://c4model.com/**](https://c4model.com/)

[**https://developer.ibm.com/patterns/online-order-processing-system-during-pandemic/**](https://developer.ibm.com/patterns/online-order-processing-system-during-pandemic/) [**https://www.ibm.com/cloud/architecture**](https://www.ibm.com/cloud/architecture) [**https://aws.amazon.com/architecture**](https://aws.amazon.com/architecture)

[**https://medium.com/the-internal-startup/how-to-draw-useful-technical-architecture-diagrams-2d20c9fda90d**](https://medium.com/the-internal-startup/how-to-draw-useful-technical-architecture-diagrams-2d20c9fda90d)

**Project Design Phase-II**

**Technology Stack (Architecture & Stack)**

|  |  |
| --- | --- |
| Date | 25 June 2025 |
| Team ID | LTVIP2025TMID33542 |
| Project Name | Grainpalette |
| Maximum Marks | 4 Marks |

**Technical Architecture:**

The Deliverable shall include the architectural diagram as below and the information as per the table1 & table 2

**Example: Order processing during pandemics for offline mode**

**Reference:** [**https://developer.ibm.com/patterns/ai-powered-backend-system-for-order-processing-during-pandemics/**](https://developer.ibm.com/patterns/ai-powered-backend-system-for-order-processing-during-pandemics/)

**S.No**

**Component**

**Description**

**Technology**

1.

User

Interface

How

user

interacts

with

application

e.g.

Web UI, Mobile App, Chatbot etc.

HTML,

CSS,

JavaScript

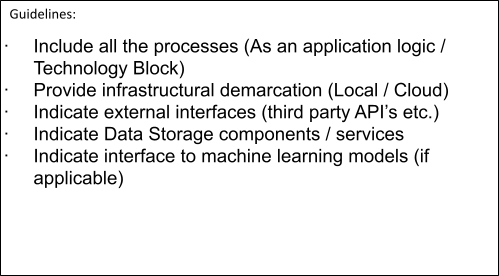
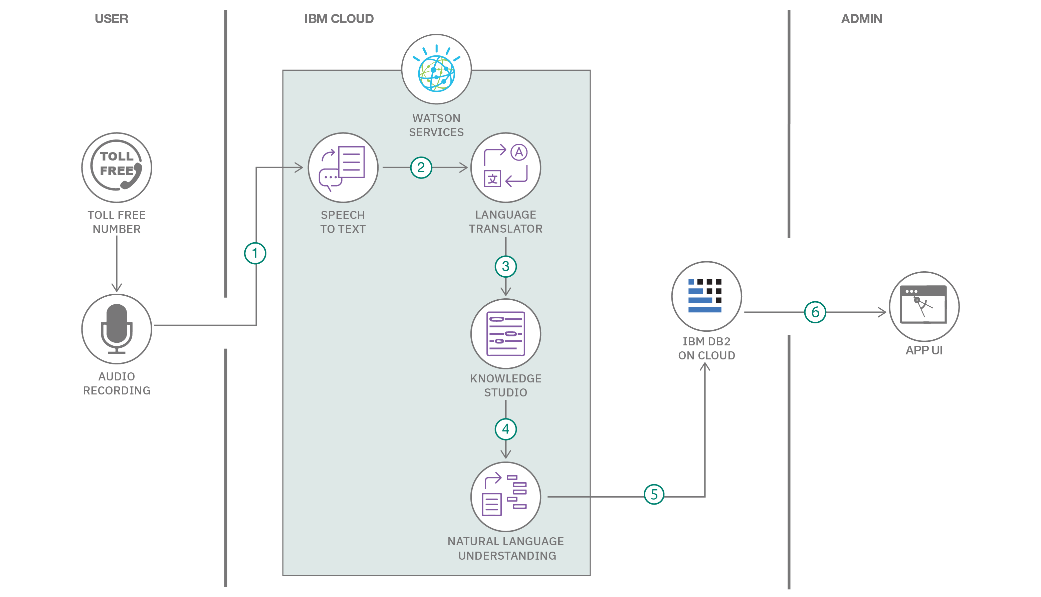
/

Angular

Js

/

React Js etc.



**Table**

**-**

**1**

**:**

**Components**

**&**

**Technologies:**

|  |  |  |  |
| --- | --- | --- | --- |
| **S.No** | **Characteristics** | **Description** | **Technology** |
| 1. | Open-Source Frameworks | List the open-source frameworks used | Technology of Opensource framework |
| 2. | Security Implementations | List all the security / access controls implemented, use of firewalls etc. | e.g. SHA-256, Encryptions, IAM Controls, OWASP etc. |
| 3. | Scalable Architecture | Justify the scalability of architecture (3 – tier, Micro-services) | Technology used |
| 4. | Availability | Justify the availability of application (e.g. use of load balancers, distributed servers etc.) | Technology used |

|  |  |  |  |
| --- | --- | --- | --- |
| 2. | Application Logic-1 | Logic for a process in the application | Java / Python |
| 3. | Application Logic-2 | Logic for a process in the application | IBM Watson STT service |
| 4. | Application Logic-3 | Logic for a process in the application | IBM Watson Assistant |
| 5. | Database | Data Type, Configurations etc. | MySQL, NoSQL, etc. |
| 6. | Cloud Database | Database Service on Cloud | IBM DB2, IBM Cloudant etc. |
| 7. | File Storage | File storage requirements | IBM Block Storage or Other Storage Service or Local Filesystem |
| 8. | External API-1 | Purpose of External API used in the application | IBM Weather API, etc. |
| 9. | External API-2 | Purpose of External API used in the application | Aadhar API, etc. |
| 10. | Machine Learning Model | Purpose of Machine Learning Model | Object Recognition Model, etc. |
| 11. | Infrastructure (Server / Cloud) | Application Deployment on Local System / Cloud  Local Server Configuration:  Cloud Server Configuration : | Local, Cloud Foundry, Kubernetes, etc. |

**Table-2: Application Characteristics:**

|  |  |  |  |
| --- | --- | --- | --- |
| **S.No** | **Characteristics** | **Description** | **Technology** |
| 5. | Performance | Design consideration for the performance of the application (number of requests per sec, use of Cache, use of CDN’s) etc. | Technology used |

**References:**

[**https://c4model.com/**](https://c4model.com/)

[**https://developer.ibm.com/patterns/online-order-processing-system-during-pandemic/**](https://developer.ibm.com/patterns/online-order-processing-system-during-pandemic/) [**https://www.ibm.com/cloud/architecture**](https://www.ibm.com/cloud/architecture) [**https://aws.amazon.com/architecture**](https://aws.amazon.com/architecture)

[**https://medium.com/the-internal-startup/how-to-draw-useful-technical-architecture-diagrams-2d20c9fda90d**](https://medium.com/the-internal-startup/how-to-draw-useful-technical-architecture-diagrams-2d20c9fda90d)

**Project Report Format**

1. **INTRODUCTION**
   1. Project Overview
   2. Purpose
2. **IDEATION PHASE**
   1. Problem Statement
   2. Empathy Map Canvas
   3. Brainstorming
3. **REQUIREMENT ANALYSIS**
   1. Customer Journey map
   2. Solution Requirement
   3. Data Flow Diagram
   4. Technology Stack
4. **PROJECT DESIGN** 
   1. Problem Solution Fit
   2. Proposed Solution
   3. Solution Architecture
5. **PROJECT PLANNING & SCHEDULING** 
   1. Project Planning
6. **FUNCTIONAL AND PERFORMANCE TESTING** 
   1. Performance Testing
7. **RESULTS** 
   1. Output Screenshots
8. **ADVANTAGES & DISADVANTAGES**
9. **CONCLUSION**
10. **FUTURE SCOPE 11. APPENDIX**

Source Code(if any)

Dataset Link

GitHub & Project Demo Link

## Full Stack Development with MERN Project Documentation format

### 1. Introduction

* **Project Title:** [Your Project Title]
* **Team Members:** List team members and their roles.

### 2. Project Overview

 **Purpose:** Briefly describe the purpose and goals of the project.  **Features:** Highlight key features and functionalities.

### 3. Architecture

* **Frontend:** Describe the frontend architecture using React.
* **Backend:** Outline the backend architecture using Node.js and Express.js.  **Database:** Detail the database schema and interactions with MongoDB.

### 4. Setup Instructions

* **Prerequisites:** List software dependencies (e.g., Node.js, MongoDB).
* **Installation:** Step-by-step guide to clone, install dependencies, and set up the environment variables.

### 5. Folder Structure

* **Client:** Describe the structure of the React frontend.
* **Server:** Explain the organization of the Node.js backend.

### 6. Running the Application

 Provide commands to start the frontend and backend servers locally.

o **Frontend:** npm start in the client directory. o **Backend:** npm start in the server directory.

### 7. API Documentation

* Document all endpoints exposed by the backend.
* Include request methods, parameters, and example responses.

### 8. Authentication

 Explain how authentication and authorization are handled in the project.  Include details about tokens, sessions, or any other methods used.

1. **User Interface** 
   * Provide screenshots or GIFs showcasing different UI features.

1. **Testing** 
   * Describe the testing strategy and tools used.

1. **Screenshots or Demo** 
   * Provide screenshots or a link to a demo to showcase the application.

1. **Known Issues** 
   * Document any known bugs or issues that users or developers should be aware of.

### 13. Future Enhancements

 Outline potential future features or improvements that could be made to the project.