MAGIC CELL – A STUDENT SERVICE CELL

Keywords: Web development, College student service cell, Student concerns, College management, User interface, Student issues, Authentication system

Abstract. This report presents the development of a web-based platform for a college student service cell called "MAGIC CELL". The platform acts as an interface for students to submit their issues to the appropriate authority within the college. The student service cell then contacts the student regarding their issue and communicates with the college management to resolve it.

The platform has been developed using **HTML**, **CSS**, **JavaScript** for front-end, **PHP**, and **MySQL** for back-end and database. The report provides a technical overview of the system, including the architecture, hardware and software requirements, and implementation details. It also provides an overview of the deployment and maintenance of the project. Finally, the report concludes with a summary of the project's challenges faced, and lessons learned.

Introduction

College life is a busy and dynamic period that requires students to be organized and proactive in addressing their needs and concerns. One of the challenges faced by students is the lack of a centralized platform to report their issues and track their requests. To address this challenge, the college has established a student service cell called MAGIC CELL, which acts as an interface between students and the college management.

This project focuses on developing a web-based platform to support the activities of the MAGIC CELL. The platform provides a user-friendly and accessible solution for students to submit their issues and monitor the progress of their requests. The platform has been developed using HTML, CSS, JavaScript for front-end, PHP, and MySQL for back-end and database.

This report provides a comprehensive overview of the project, including its background, objectives, technical aspects, design, implementation, testing, deployment, and maintenance. It concludes with a summary of the project's achievements, challenges faced, and lessons learned.

Problems to be addressed

- Lack of a centralized platform: The project addresses the issue of students having no centralized platform to report their issues and track their requests.

- Communication gap: The project helps to bridge the communication gap between students and the college management by providing an interface for students to report their concerns.

- Inefficient resolution process: The project streamlines the process of resolving student issues by allowing them to report their concerns directly to the appropriate authority.
- Lack of transparency: The project provides a transparent and accessible solution for students to monitor the progress of their requests, ensuring accountability and transparency in the resolution process.

Goals/Objectives

This part of the project plan helps team members move toward project execution smoothly. Make sure to include:

- To provide a centralized platform for students to report their issues and track their requests.
- To bridge the communication gap between students and the college management.
- To streamline the process of resolving student issues.
- To ensure accountability and transparency in the resolution process.
- To provide a user-friendly and accessible solution for students to submit their concerns.
- To support the activities of the MAGIC CELL.
- To provide a comprehensive and efficient platform for the college management to manage and resolve student issues.
- To ensure the platform is scalable, secure, and easily maintainable.

Scope of Work

This project entails research, surveys, training and knowledge of web development.

- Requirements gathering and analysis: This includes understanding the needs and requirements of the college management and students, and defining the functional and technical requirements of the platform.
- Design and prototyping: This include creating a detailed design for the platform, including its user interface, database schema, and architecture. A prototype will also be developed to demonstrate the platform's functionality.
- Front-end development: This includes developing the user interface of the platform using HTML, CSS, and JavaScript, ensuring a responsive and user-friendly design.
- Back-end development: This includes developing the back-end of the platform using PHP and MySQL, implementing the functionality for submitting and tracking requests, and managing the database.
- Testing and quality assurance: This includes testing the platform to ensure it meets the functional and technical requirements, and fixing any identified issues.
- Deployment and maintenance: This include deploying the platform on a web server, providing user training and support, and ongoing maintenance and improvement of the platform.
- Documentation: This includes preparing a detailed project report, documenting the design, implementation, testing, deployment, and maintenance of the platform.

Methodology

The Waterfall model is a widely used sequential development methodology that follows a linear and sequential approach to software development. In this model, each stage of development is completed before moving on to the next stage, and there is minimal overlap between stages. The steps involved in the Waterfall model are:

- Requirements gathering and analysis: During this phase, the requirements of the platform are gathered and analyzed. This includes understanding the needs and requirements of the college management and students, and defining the functional and technical requirements of the platform. A detailed requirements specification is created during this phase, which serves as the basis for the rest of the development process.

- Design: During this phase, the design of the platform is created, including the user interface, database schema, and architecture. A prototype of the platform is also created, which serves as a visual representation of the platform and helps to validate the design. The design phase includes a detailed specification of the user interface and database design, as well as a description of the algorithms and processes used in the back-end.

- Implementation: During this phase, the front-end and back-end of the platform are developed using HTML, CSS, JavaScript, PHP, and MySQL. The implementation phase involves writing code and building the platform, following the design specifications created in the previous phase.

- Testing: During this phase, the platform is tested to ensure it meets the functional and technical requirements. This includes unit testing, integration testing, and system testing, as well as testing the platform's security and performance. Any identified issues are fixed during this phase.

- Deployment: During this phase, the platform is deployed on a web server and made available for use. This phase also involves configuring the platform for production use and setting up any required infrastructure and hardware.

- Maintenance: During this phase, ongoing maintenance and improvement of the platform are performed to ensure it continues to meet the needs of the college and its students. This includes fixing bugs, adding new features, and ensuring the platform remains secure and reliable.

- The Waterfall model is well-suited to this project as it provides a clear and structured approach to software development, with well-defined stages, and a clear progression from one stage to the next. This allows for better planning, testing, and quality assurance, and reduces the risk of delays and rework.

Design and Implementation

The design of the website was created to be user-friendly and visually appealing to the students and college management. The user interface was designed with a focus on simplicity, so that students and management could easily navigate and use the platform. The design was created using HTML, CSS, and JavaScript, which were used to create the user interface and define the visual style of the website.

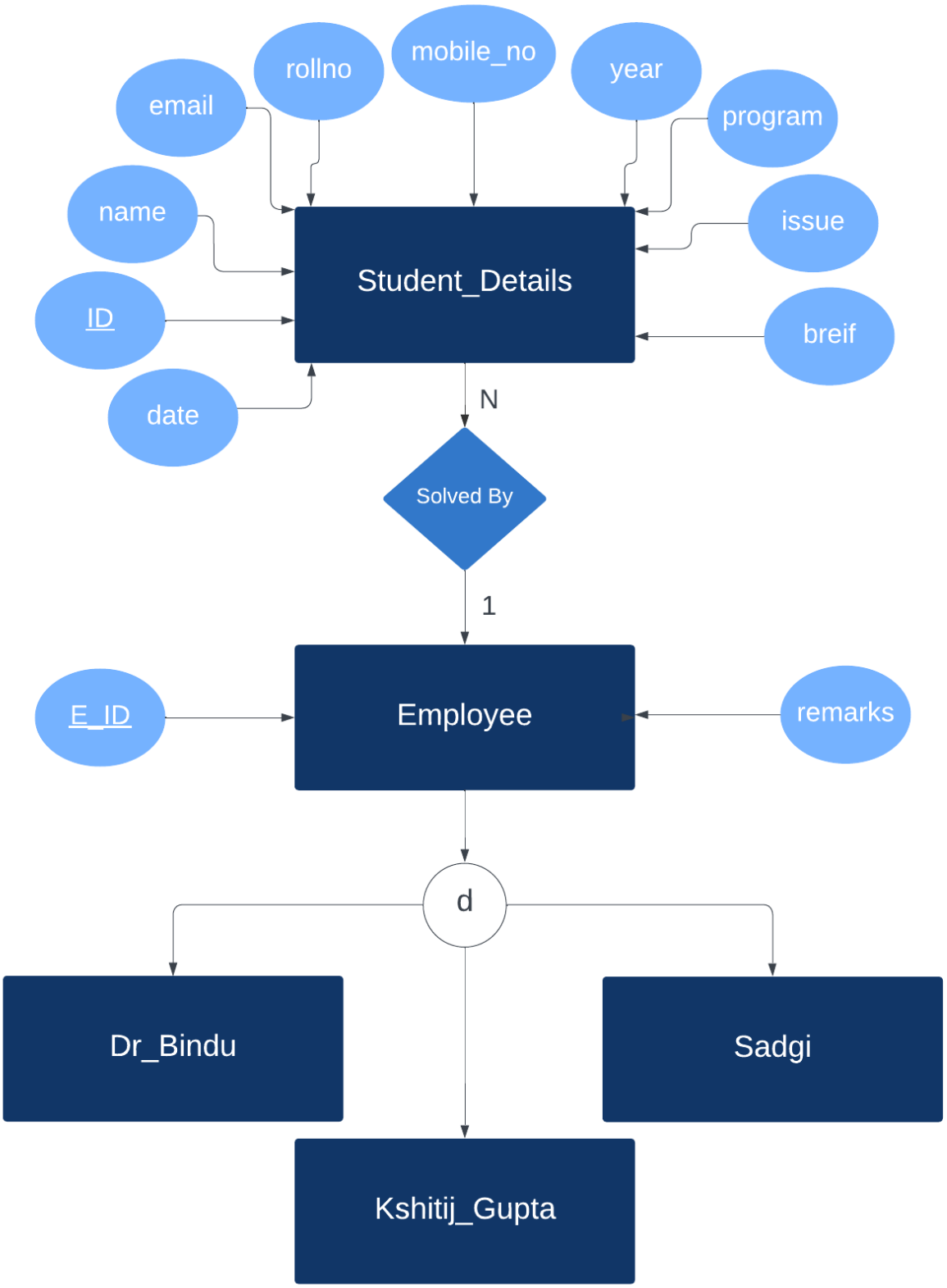
The back-end of the platform was implemented using PHP and MySQL, which were used to create the database and store the data. The back-end was designed to handle the submission of issues by students and the management, as well as the tracking of the progress of each issue.

The implementation of the platform followed the design created in the design phase, and was developed using the Waterfall model methodology. The platform was built in stages, with each stage being completed before moving on to the next. This allowed for a structured and well-planned approach to development, and reduced the risk of delays and rework.

The platform was thoroughly tested during the testing phase to ensure it met the functional and technical requirements. Any identified issues were fixed during this phase.

The platform was designed and implemented with scalability in mind, so that it could easily be expanded in the future to meet the changing needs of the college and its students. Ongoing maintenance and improvement of the platform are performed to ensure it continues to meet the needs of the college and its students, and to ensure it remains secure and reliable.

ER DIAGRAM :


mobile_no
rollno
year
email
program
name
issue
Student_Details
ID
breif
date
N
Solved By
1
E_ID
Employee
remarks
d
Dr_Bindu
Sadgi
Kshitij_Gupta

Conclusion

In conclusion, the "MAGIC CELL" web development project aimed to build a platform for college students to raise their concerns and track the progress of their requests. The project utilized HTML, CSS, JavaScript for the frontend, PHP and MySQL for the backend and database. The platform acts as an interface for students to submit their issues to the appropriate authority for resolution.

However, there are still some future works that need to be addressed to further improve the platform's functionality. These include:

Implementation of a single point of access for students to raise their concerns and track the progress of their requests.

Integration of notifications and alerts to inform students of any updates on their requests.
Adding more user-friendly features to enhance the user experience.
Improving the security of the platform by implementing proper authentication and authorization measures.

Regular maintenance and updates to ensure the platform remains up-to-date and functional.

Overall, the "MAGIC CELL" web development project was a successful endeavor in providing a platform for college students to communicate their concerns and issues to the appropriate authority. However, there is still room for improvement and future work required to make it more user-friendly and secure.

References

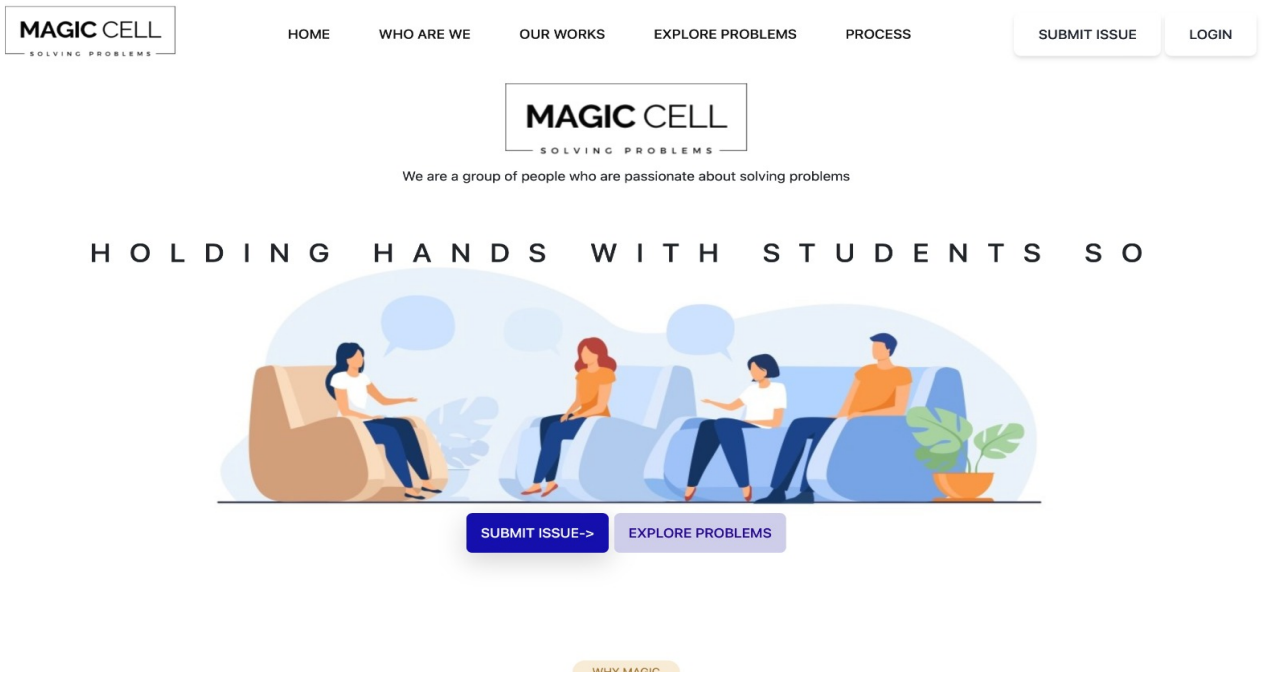
[1] Design ideas from https://www.amaxaimpact.org/

Reference to a book:

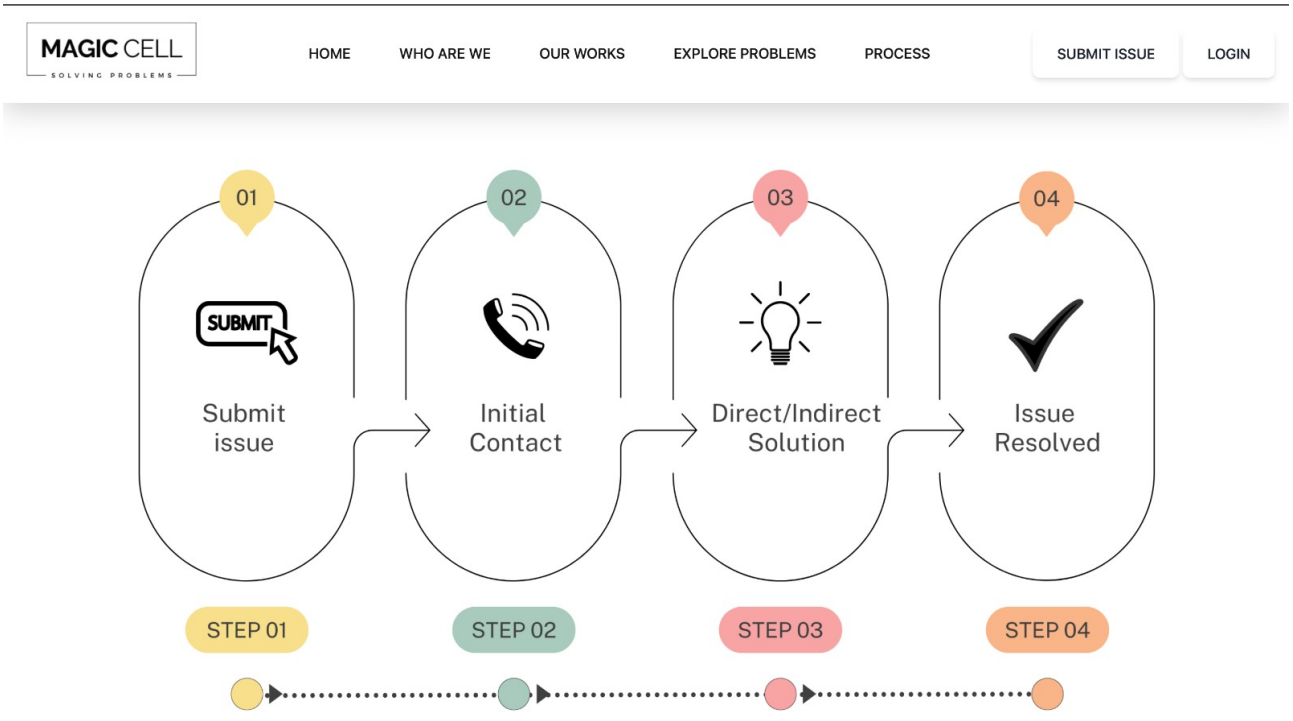
[2] Vector/SVG ideas from https://www.freepik.com/

[3] Code help from https://github.com/akashyap2013/Books-bootstrap-website

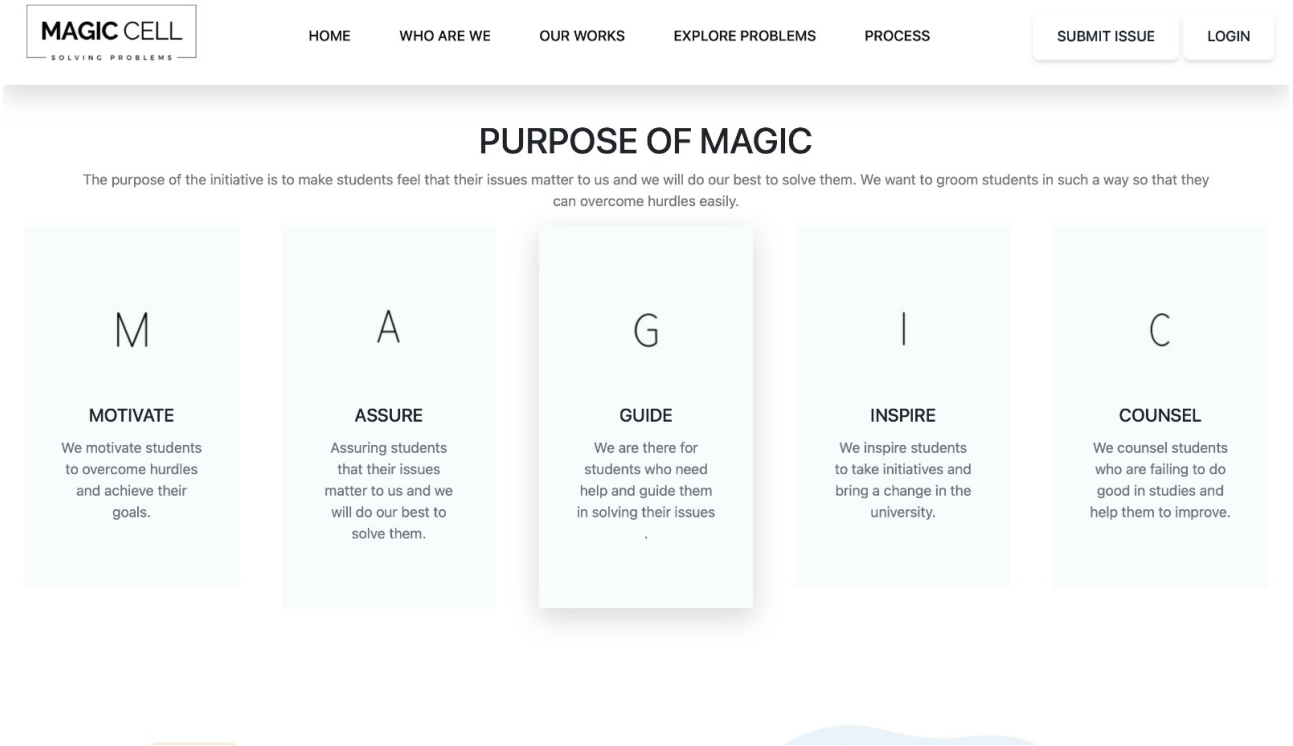
The Home Page:



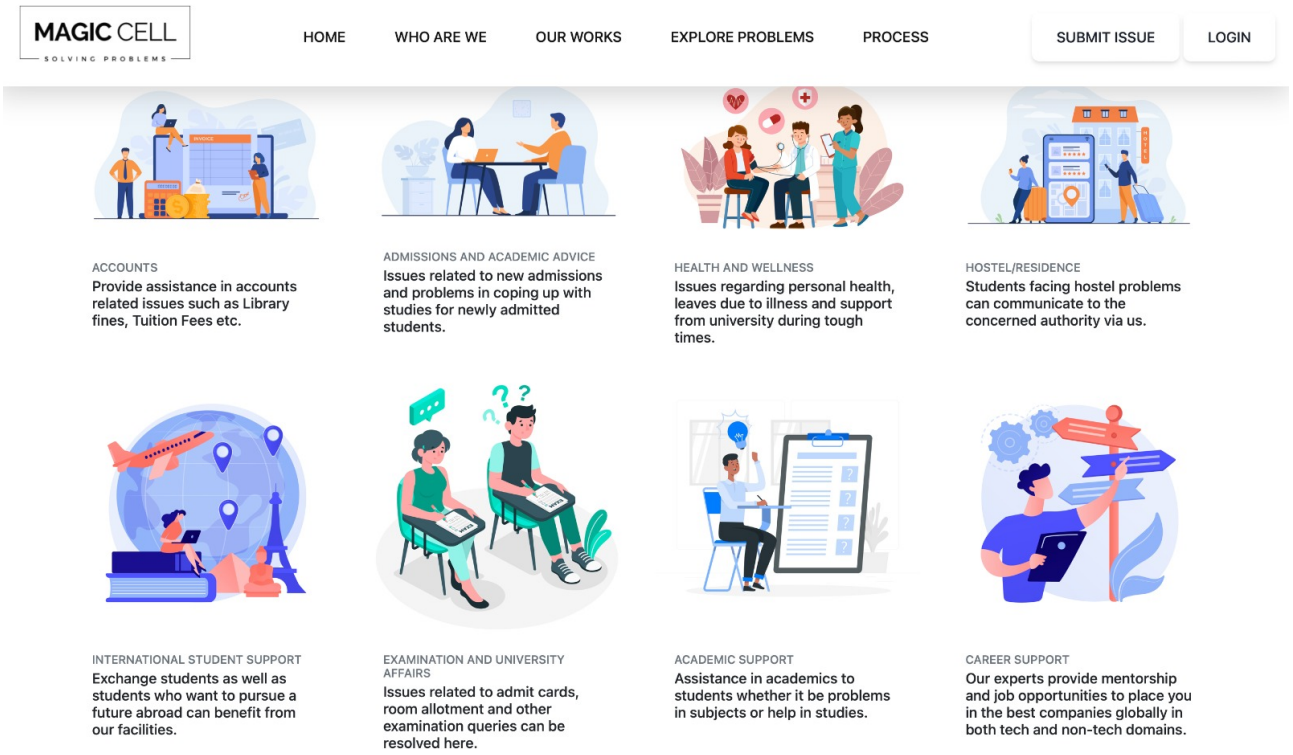
The process :



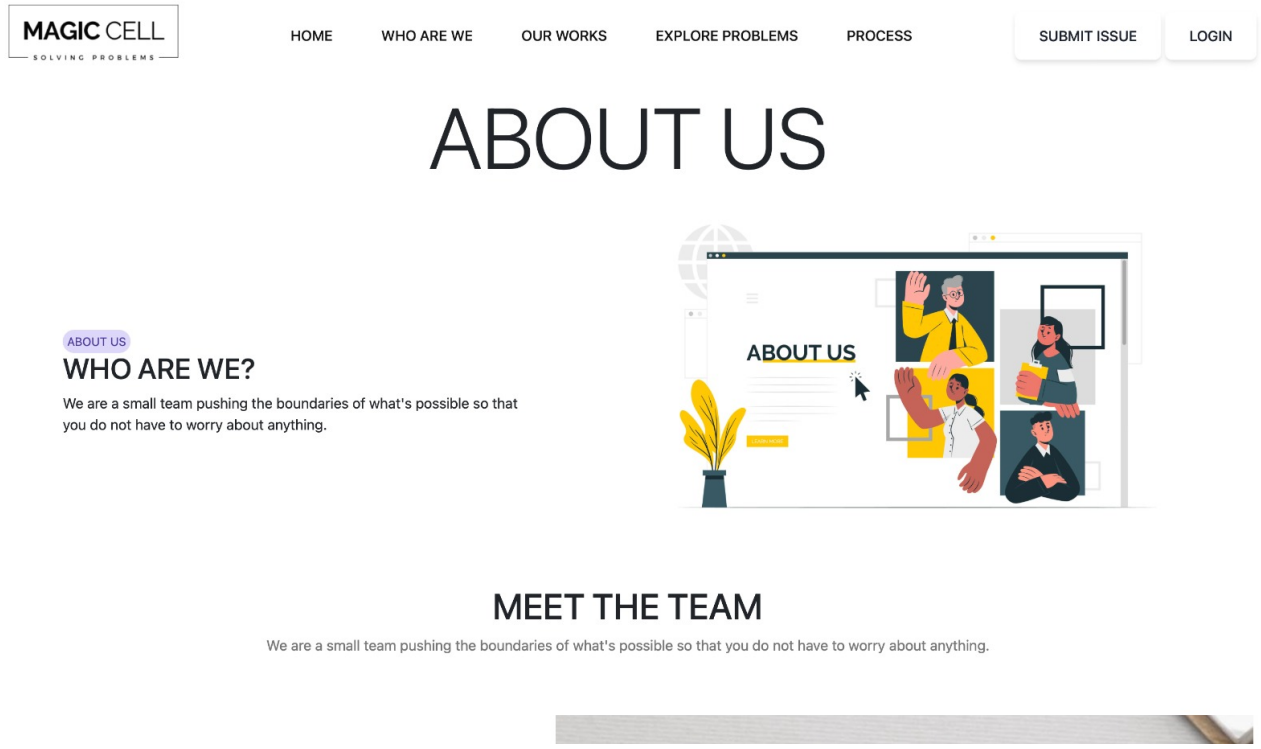
Purpose of magic :



Explore problems page :



Who are we ? :



Our works :