

OVERVIEW

HackerEarth is back with another Full-stack development hackathon. Welcome to StackHack 2.0, inviting all developers and hackathon enthusiasts to come up with fantastic ideas to build minimal viable products from scratch. Code your way from the backend to the frontend and design products to inspire!

It's true that being a specialist in one's field and gaining mastery of a particular aspect of technology has distinct advantages, but as technology rapidly expands and evolves, industries are constantly changing. With this change comes the need for skills branching out to multiple technologies and the ability to build end to end products to potentially transform the way people live around the world.

This hackathon is open to all developers who have an eye for design and a passion for code. Participants can either hack individually or as a team of four. Take advantage of this opportunity to learn with other teams, showcase your talent, and compete for prizes!

What are we looking for in the submission?

We are looking for prototypes that are innovative, technologically advanced and unique. Most importantly, we are looking for applications that help fuel innovation, industry growth and economy. Build a web applications/ products that help everyday business manage employee information better and allows employees to save time and order meals with ease at the office cafeteria hassle-free. You can make a submission for either of the theme mentioned below.

THEMES



Human Resource management

[Objective] - Develop a pseudo full-stack web-app for HR management

[Minimum task] - Employee Information Management

On Front End:

- Employee Records and Database - Implement a feature to add employees and their details (personal, role, team, salary, etc).
- Attendance Management - Implement a feature to allow employees to apply/mark their leaves.

On Back end:

- **Implement the backend in one of the desired Tech-Stacks provided below.**
- Backend should store all the tasks data received from the Frontend, in the Database.
- Implement a Database in the Backend which should store all this structured data.
- Data sharing between Frontend and Backend should be in JSON format rendered over REST APIs.
- Zip all your Source Code, Screenshots, Deployment Instructions and Upload.

Deployment :

- Deploy your Fullstack App on any Cloud platform or service. E.g. Heroku, python anywhere, or AWS Elastic Beanstalk, GCP or Microsoft Azure or any free alternative or any other way. (This deployment link must be active for at least 2 weeks from Submission Date.)
- Provide the Deployment URL under “demo link” field.

[Intermediate task]

- Implement a feature for Payroll Management and requesting for Loans and Bonuses.
- Allow admins to mark leave-requests as “approved” or “rejected”.
- Store the relevant flags in the database.
- Allow users to check company docs like holidays, policies, FAQ, etc.
- Implement these features for an End-to-End stack. Implementation will go on the Frontend as well as Backend.

[Plus points] (experienced / wizard)

- Implement the Signup and Login/Logout functionality. You have to create a user-auth schema in the database.
- Implement a feature to Search and Filter employees based on team (BU), org. hierarchy and titles etc, and a combination of at least two or more.

Note : If you complete any or all of the above mentioned levels please do submit.

Ideal Stack : MEAN/MERN Stack (Preferred), you can also submit in any technology of your choice.



Office Cafeteria App

[Objective] - Develop a pseudo full-stack app for Office cafeteria

[Minimum Requirement] (entry level)

On Front End:

- **Create a form on the UI which captures :**
 - Fields for Data Capture (All fields are mandatory)
 - Full Name
 - Organization name
 - Employee ID no.
 - Mobile No.
- **E-Mail**
- Upload ID Card (Formats: png, jpeg)
- Provide a preview Screen which should :
 - List all the fields as above.
 - Display ID card in the preview.
- On Submission :
 - Registration ID has to be generated and displayed on Success Screen.
 - Store all the information captured in a local database
 - Registration Date has to be generated as system date and get stored in Database

On Back end:

- **Implement the backend in one of the desired Tech Stacks provided below.**
- Backend should have database bindings to store the event registration info received from the Front End.
- The backend should be able to receive from and render to Front End all the event information, also store in and fetch from the database.
- Submit Screenshot, Source code & Deployment Instructions.

Deployment :

- Deploy your Fullstack App on any Cloud platform or service. E.g. Heroku, pythonanywhere, or AWS Elastic Beanstalk, GCP or Microsoft Azure or any free alternative or any other way. (This deployment link must be active for at least 2 weeks from Submission Date.)
- Provide the Deployment URL under “demo link” field.

[Intermediate task]

- Display a Chart (food menu) detailing food items with price.
- Option to choose mode of payment

Plus Point (intermediate)

- Along with everything asked in Minimum Requirement :
 - Sign-up/Login functionality.
 - Set time for take-away of order
 - Documented, well structured beautified code.
-

Note : If you complete any or all of the above mentioned levels please do submit.

Ideal Stack : MEAN/MERN Stack (Preferred), you can also submit in any technology of your choice.