

PROBLEM SET FOR APP DEVELOPMENT

- This is team based event (a team can have 2 or 3 members).
- Select any one problem statement from the 6 questions given below.
- Candidates can submit problems 1-4 in either PHP or JSP (along with SQL or Oracle DB).
- Problems 5 and 6 are to be implemented in only Android (API 17+).
- Each team will have to send a SRS in PDF format to **effulgenceappdev@outlook.com** with the subject "SRS (team name) (platform)" without parentheses or inverted commas. Example- if a team named Legalized is sending their SRS on the problem on PHP, they have to send it to the aforementioned email ID with the subject **SRS Legalized PHP**.
The email should also contain the name, email ID, college and contact information for all the members in the team.
Only one member of the team can send all the above information.
- Each team is required to submit a working application for only one of the problems given below. However if a team wants to submit entries for more than one problem, they can do so by submitting different SRS documents and source code zip files in separate emails (in the format mentioned in the rule).
- Contact-

 effulgenceappdev@outlook.com

 8005381618

1. Online polling (PHP or JSP)

Create an online polling / voting website where out of a set of 5 choices, a user has to poll or select one.

- The home page of the website should be a registration form where new users can give in their details- Name, age, gender, city, state, pin-code.
- After registration, the users should be directed to a polling page where 5 choices with random text (or text- Option1, Option2, Option3, Option4, Option5) is displayed. The user can poll or select only one of these (using radio buttons) and click submit.
- After submission, he should be redirected back to home page.
- The home page should also have a link (button) for admin login. On clicking the link for admin login the a sign-in page should come which prompts the admin to log in with his username and password (there is no requirement for a form or another page for admin sign-up, the username and password for admin should be fed into database directly).
- After logging in, the admin should be able to view the result which will display in integers how many users have polled for a particular option (out of the 5 options displayed during polling). He should also have links (buttons) to sort the result based on gender, state, age-group.

2. Online Diary (PHP or JSP)

Create an online diary based note taking website which can store notes of registered users and display them later.

- The home page should contain links for sign-in and sign-up.
- The sign-up page can have details such as names, age, gender, email-ID, password.
- The sign-in page should have email text box and password box to log into user's account.
- On logging-in, the user should be shown a text box where he/she can enter texts or notes (without word limit) and press the save button to save it.
- He/she should also be shown a link called "Archived" to show the user's previous notes along with date and time when the note was saved.
- The password used during sign-up should be more than 8 characters and contain a mix of uppercase, lowercase, numbers and special symbols. The password should be stored as MD5 hash in the database.

3. File upload and download (PHP or JSP)

Create a website (web based portal) where a user can log-in and then store his files by uploading.

- The home page should display links of sign-in and sign-up.
 - The sign-up page can have details such as names, age, gender, email-ID, password.
 - The sign-in page should have email text box and password box to log into user's account.
 - On logging-in the user should be shown his previously uploaded files or data and a button to add more files to his account.
 - On clicking any previously uploaded file or file name, the file should start downloading.
-

4. Web crawler website (PHP or JSP)

Create a website where a user can enter a URL and a single word, to count the number of times the given word appears in the page linked to the URL.

- The home page should show only two text boxes, one should be used for entering the URL and the other should be used for entering the single search word.
- The crawler used in the background should count the number of times the word appears in the page directed by the URL and in the links contained on the page (but not beyond) i.e. the web crawler logic should scan the page of the URL and the pages contained in the hyperlink (if any) shown in the page but not more.
- The result should be displayed as an integer on the home page itself.

5. Expression evaluator (Android)

Create an Android app that contains a text section for entering a mathematical expression and a button “Show result” that creates a new activity to show the result of the expression.

- The text box should take in an expression of the form such as $3+4*2-3$ etc and give the result in the new activity.
 - The operators allowed in the expression are $+ - * /$
 - Candidates can implement parentheses checking in the expression such as $5+4*(8/2-1)+7$ (for which they will be awarded extra credit). However this is not a mandatory requirement.
-

6. RSS news reader (Android)

Create an Android app that shows the content heading when a RSS feed is given to it.

- The app should only show a text box (where a RSS feed link of the form <http://feeds.mashable.com/Mashable> can be entered) and a “show content” button which will show only the headings of the article fetched from the feed link. On clicking the heading, the article should open up for viewing.